SCADAPack

ISaGRAF 3.55 to RemoteConnect Migration Guide

Version: 4.1.1

Date: December 2022

Document Number: 250543

Table of Contents

1 Legal Information

The information provided in this documentation contains general descriptions and/or technical characteristics of the performance of the products contained herein. This documentation is not intended as a substitute for and is not to be used for determining suitability or reliability of these products for specific user applications. It is the duty of any such user or integrator to perform the appropriate and complete risk analysis, evaluation and testing of the products with respect to the relevant specific application or use thereof. Neither Schneider Electric nor any of its affiliates or subsidiaries shall be responsible or liable for misuse of the information contained herein. If you have any suggestions for improvements or amendments or have found errors in this publication, please notify us.

All pertinent state, regional, and local safety regulations must be observed when installing and using this product. For reasons of safety and to help ensure compliance with documented system data, only the manufacturer should perform repairs to components.

This document contains standardized industry terms that some customers might find insensitive or offensive. These terms do not reflect the official policy or position of Schneider Electric.

Trademarks

Address

Schneider Electric

35 rue Joseph Monier
92500 Rueil-Malmaison, France
Email: RemoteOperations@se.com
www.se.com

2 Technical Support

Questions and requests related specified sections of this document can be directed to one of the following support centers.

Support Required	Contact Information
Specific issues related to ISaGRAF, Workbench, or RemoteConnect Sections 1, 2, 3, 4, 5, 7, 11.2, and 12	supportRO@se.com • Toll free within North America: 1-888-226-6876 • Direct Worldwide: +1-613-591-1943
EcoStruxure Control Engineering website Sections 11 and 11.1	global-emro-support@schneider-electric.com
General issues not covered by the above teams Sections 6, 8, 9, and 10	RemoteOperations@se.com

3 Safety Information

Important information

Read these instructions carefully, and look at the equipment to become familiar with the device before trying to install, operate, or maintain it. The following special messages may appear throughout this documentation or on the equipment to warn of potential hazards or to call attention to information that clarifies or simplifies a procedure.

The addition of this symbol to a Danger or Warning safety label indicates that an electrical hazard exists, which will result in personal injury if the instructions are not followed.

This is the safety alert symbol. It is used to alert you to potential personal injury hazards. Obey all safety messages that follow this symbol to avoid possible injury or death.

⚠ DANGER

DANGER indicates a hazardous situation which, if not avoided, **will result in** death or serious injury.

⚠ WARNING

WARNING indicates a hazardous situation which, if not avoided, **can result in** death or serious injury.

⚠ CAUTION

CAUTION indicates a hazardous situation which, if not avoided, **can result in** minor or moderate injury.

NOTICE

NOTICE is used to address practices not related to physical injury.

Please note

Electrical equipment should be installed, operated, serviced, and maintained only by qualified personnel. No responsibility is assumed by Schneider Electric for any consequences arising out of the use of this material.

A qualified person is one who has skills and knowledge related to the construction, installation, and operation of electrical equipment and has received safety training to recognize and avoid the hazards involved.

Before you begin

Do not use this product on machinery lacking effective point-of-operation guarding. Lack of effective point-of-operation guarding on a machine can result in serious injury to the operator of that machine.

⚠ WARNING

EQUIPMENT OPERATION HAZARD

- Verify that all installation and setup procedures have been completed.
- Before operational tests are performed, remove all blocks or other temporary holding means used for shipment from all component devices.
- Remove tools, meters, and debris from equipment.

Failure to follow these instructions can result in death or serious injury.

Follow all start-up tests recommended in the equipment documentation. Store all equipment documentation for future reference.

Test all software in both simulated and real environments.

Verify that the completed system is free from all short circuits and grounds, except those grounds installed according to local regulations (according to the National Electrical Code in the U.S.A, for instance). If high-potential voltage testing is necessary, follow recommendations in equipment documentation to help prevent accidental equipment damage.

Operation and adjustments

The following precautions prevail:

- Regardless of the care exercised in the design and manufacture of equipment or in the selection and ratings of components, there are hazards that can be encountered if such equipment is improperly operated.
- It is sometimes possible to misadjust the equipment and thus produce unsatisfactory or unsafe operation. Always use the manufacturer's instructions as a guide for functional adjustments. Personnel who have access to these adjustments should be familiar with the equipment manufacturer's instructions and the machinery used with the electrical equipment.
- Only those operational adjustments actually required by the operator should be accessible to the operator. Access to other controls should be restricted to prevent unauthorized changes in operating characteristics.

Acceptable use

⚠ WARNING

UNACCEPTABLE USE

Do not use SCADAPacks or I/O modules as an integral part of a safety system. These devices are not safety products.

Failure to follow these instructions can result in death or serious injury.

⚠ CAUTION

EQUIPMENT OPERATION HAZARD

When devices are used for applications with technical safety requirements, the relevant instructions must be followed.

Use only Schneider Electric software or approved software with Schneider Electric hardware products.

Failure to follow these instructions can result in minor or moderate injury.

4 About the Book

Audience

⚠ WARNING

UNINTENDED EQUIPMENT OPERATION

The application of this product requires expertise in the design and programming of control systems. Only persons with such expertise are allowed to program, install, alter, and apply this product.

Follow all local and national safety codes and standards.

Failure to follow these instructions can result in death or serious injury.

This manual is written for people who are responsible for the initial set up and configuration of a SCADAPack x70. These individuals should be familiar with IEC 61131-1, telemetry communications, and the operation of RTUs.

These persons are typically:

- Systems Engineers
- Commissioning Engineers
- Automation Engineers

Document scope

This manual describes:

- How to access an automation project created in ISaGRAF 3.55 for a SCADAPack
- How to manually convert an ISaGRAF 3.55 project to a SCADAPack Workbench project including what functionality cannot be directly converted
- How to automatically convert a SCADAPack Workbench project to a SCADAPack RemoteConnect project including the manual steps required prior to commissioning the project.

Validity note

The information described herein applies to ISaGRAF version 3.55, SCADAPack Workbench version 6.6.8, and SCADAPack RemoteConnect configuration software version 2.7.3.

Related documents

- SCADAPack Workbench Manuals
- SCADAPack E Documentation Set
- SCADAPack x70 Documentation Set

5 Cybersecurity

Cybersecurity is a branch of network administration that addresses attacks on or by computer systems and through computer networks that can result in accidental or intentional disruptions. The objective of cybersecurity is to help provide increased levels of protection for information and physical assets from theft, corruption, misuse, or accidents while maintaining access for intended users.

No single cybersecurity approach is adequate. Schneider Electric recommends a defense-in-depth approach. This approach layers the network with security features, appliances, and processes. The basic components of this approach are:

- Risk assessment: A systematic security analysis of the environment and related systems.
- A security plan built on the results of the risk assessment
- A multi-phase training campaign
- Network separation and segmentation: Physical separation of the control network from other networks, and the division of the control network itself into segments and security zones.
- System Access Control: Controlling access to the system with firewalls, authentication, authorization, and other software means, and traditional physical security measures such as video surveillance, fences, locked doors and gates, and locked equipment cabinets.
- Device hardening: The process of configuring a device against communication-based threats. Device hardening measures include disabling unused network ports, password management, access control, and the disabling of all unnecessary protocols and services.
- Network monitoring and maintenance: An effective defense-in-depth campaign requires continual monitoring and system maintenance to meet the challenge of new threats as they develop.
- See Security Considerations in the SCADAPack DNP3 Security Technical Reference manual

Contact us

For more information, refer to the Schneider Electric Cybersecurity Support Portal at http://www.se.com/b2b/en/support/cybersecurity/overview.jsp.

Additional Resources

Schneider Electric Recommended Cybersecurity Best Practices
https://www.se.com/ww/en/download/document/CS-Best-Practices-2019-340/

Cybersecurity and Infrastructure Security Agency
https://www.cisa.gov/

ICS-CERT Recommended Practices
https://ics-cert.us-cert.gov/Recommended-Practices

Center for Internet Security (CIS) Top 20 Critical Security Controls
https://www.cisecurity.org/cybersecurity-best-practices

FBI Cyber Crime
https://www.fbi.gov/investigate/cyber

Guide to Industrial Control Systems (ICS) Security
https://www.nist.gov/publications/guide-industrial-control-systems-ics-security

WaterISAC Water Security Network
https://www.waterisac.org

6 Introduction

This document is intended as a guide in the conversion process of logic and configuration from SCADAPack Smart RTUs running ISaGRAF Target 3 applications to SCADAPack x70 RTUs running SCADAPack RemoteConnect configuration software and logic.

NOTE: This document presumes that you have some degree of experience with SCADAPack Workbench as well as ISaGRAF 3.55. The instructions below are not intended to take the place of a Workbench training course.

This process is applicable to ISaGRAF logic running on a standard SCADAPack (SCADAPack 32 and 300-Series) RTU and ISaGRAF Target 3 logic running on an E-Series SCADAPack (300 and 500-Series) RTU. Converting Workbench Application to SCADAPack RemoteConnect 46 also applies to SCADAPack Workbench Target 5 logic running on an E-Series SCADAPack.

This document does not discuss the hardware differences between the Standard RTUs or E-Series RTUs and the newer SCADAPack x70 RTUs in any detail, as there are far too many variants. However, some typical hardware is discussed.

As many of the steps, and their variations, are provided as possible. However, due to the manual nature of the process, not all specific configurations and variations that may be encountered are covered. This manual provides a guide to the process in general.

Schneider Electric can provide some assistance with this process, but it is not in any way guaranteed to work in any specific situation.

For technical assistance with any stage of the conversion process, see Technical Support 7.

The following is a basic summary of the process:

- ISaGRAF application imported into SCADAPack Workbench Target 3
- Target 5 application created, logic and variables copied from Target 3
- Target 5 application converted to a SCADAPack RemoteConnect application using Schneider Electric's EcoStruxure™ Control Engineering Converter tool
- Converted SCADAPack RemoteConnect application carefully reviewed and modified (as required) before bench or field testing

NOTES

- Workbench Target 3 requires the original ISaGRAF files, specifically the appli.hie file. It will not import an archived *.pia file.
- The EcoStruxure Control Engineering Converter website needs to receive the *.mdb file created by Workbench Target 5 to complete the logic conversion.

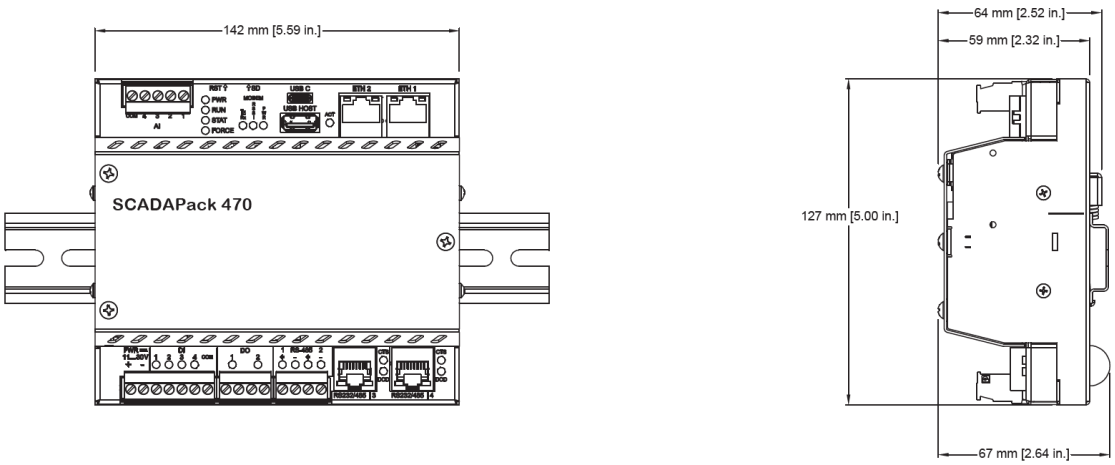
7 Hardware Considerations

See:

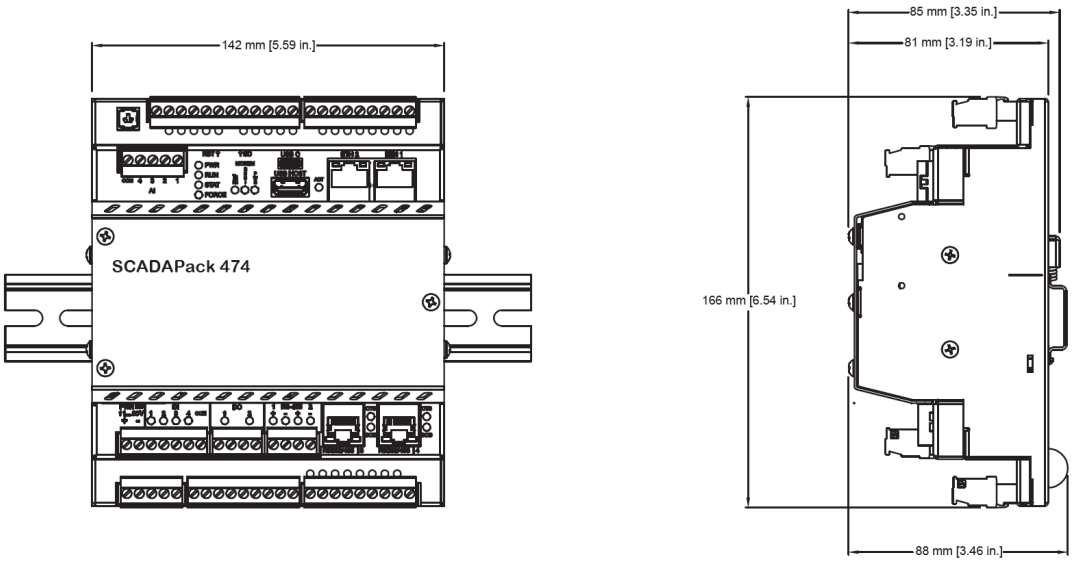
7.1 Physical Footprint

Before beginning the conversion process, the footprint of the new SCADAPack x70 model needs to be considered. As there are many Standard and E Series models, those are not documented here. User manuals and datasheets can provide that information. The enclosure space allotted to the older model needs to allow the new SCADAPack x70 RTU to be installed.

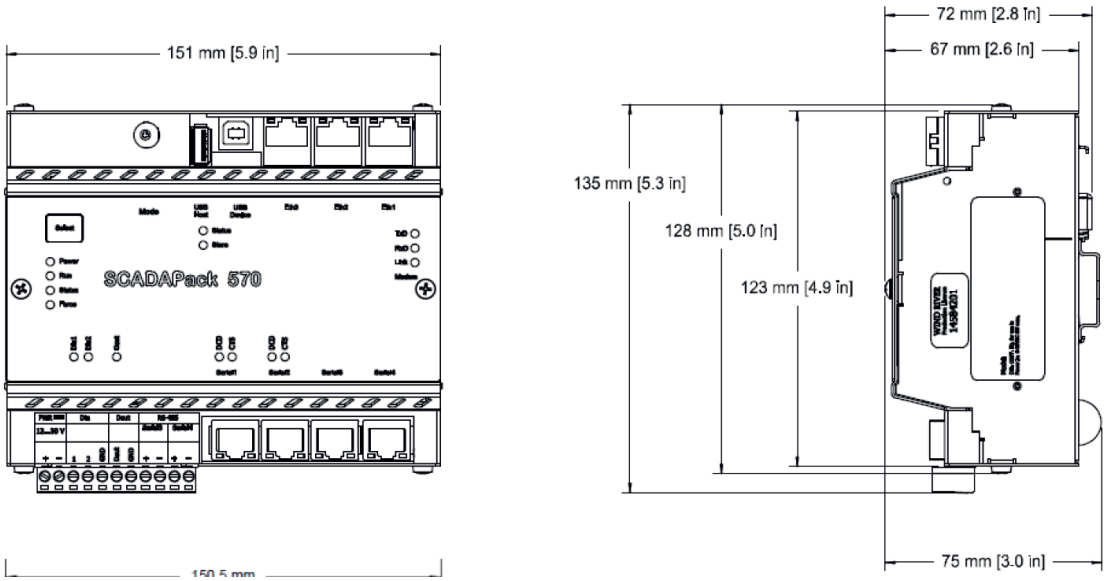
SCADAPack 470 physical dimensions and I/O layout



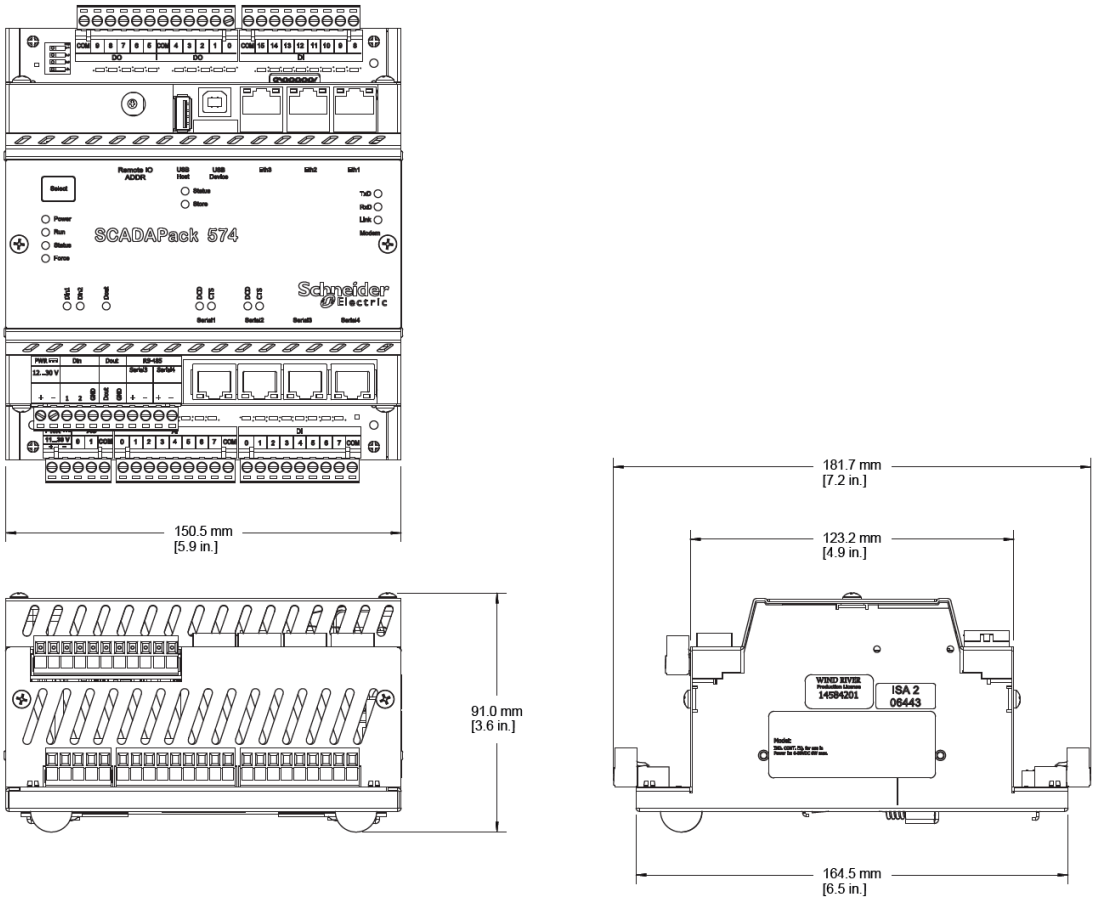
SCADAPack 474 physical dimensions and I/O



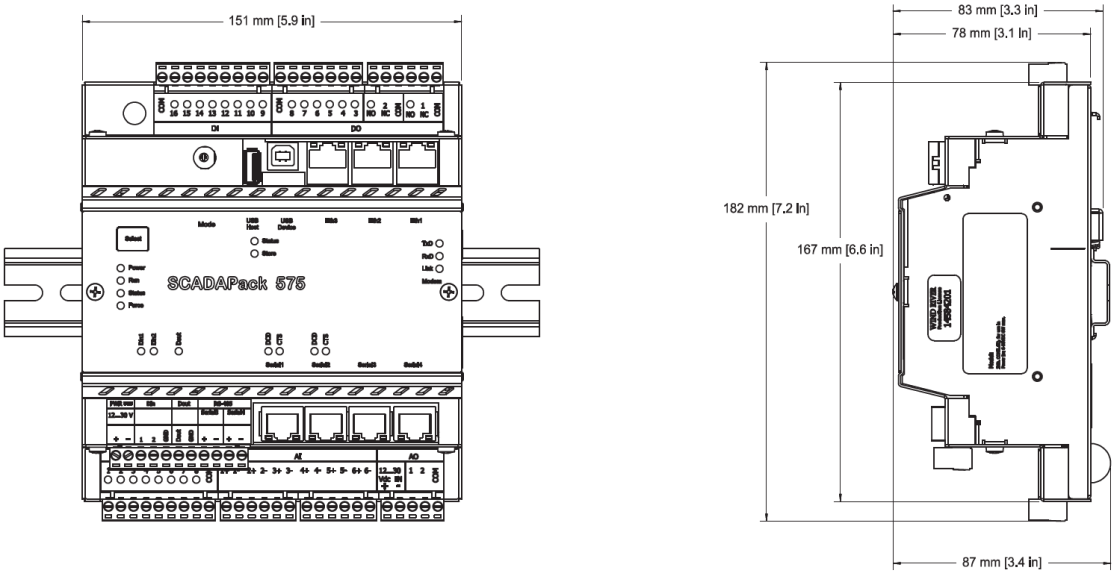
SCADAPack 570 physical dimensions and I/O layout



SCADAPack 574 physical dimensions and I/O layout



SCADAPack 575 physical dimensions and I/O layout



7.2 I/O Configuration Compatibility

The physical location, quantity, and specific characteristics of I/O need to be considered when migrating to a SCADAPack x70 RTU. The following links provide a summary, but refer to the current datasheets for full information.

See:

- Digital Inputs 18
- Digital Outputs 19
- Counter Inputs 19
- Analog Inputs 20
- Analog Outputs 20

7.2.1 Digital Inputs

Unit	I/O	Terminal Location
SCADAPack 470	4 Digital Inputs	01...04 controller board lower left
SCADAPack 474	20 Digital Inputs	01...04 controller board lower left 05...12 I/O board lower right 13...20 I/O board upper right

SCADAPack 570	2 Digital Inputs	01...02 controller board lower left
SCADAPack 574	18 Digital Inputs	01...02 controller board lower left 03...10 I/O board lower right 11...18 I/O board upper right
SCADAPack 575	18 Digital Inputs	01...02 controller board lower left 03...10 I/O board lower left 11...18 I/O board upper left

7.2.2 Digital Outputs

Unit	I/O	Terminal Location
SCADAPack 470	2 Digital Outputs	01...02 controller board lower center
SCADAPack 474	12 Digital Outputs	01...02 controller board lower center 03...12 I/O board upper left
SCADAPack 570	1 Digital Output	01 controller board lower left
SCADAPack 574	11 Digital Outputs	01 controller board lower left 02...11 I/O board upper left
SCADAPack 575	9 Digital Outputs	01 controller board lower left 02...09 I/O board upper right

7.2.3 Counter Inputs

Unit	I/O	Terminal Location
SCADAPack 470	4 Counter Inputs (shared with Dis)	01...04 controller board lower left
SCADAPack 474	12 Counter Inputs (shared with Dis)	01...04 controller board lower left 05...12 I/O board lower right
SCADAPack 570	2 Counter Inputs (Shared with Dis)	01...02 controller board lower left
SCADAPack 574	2 Counter Inputs (Shared with Dis)	01...02 controller board lower left

SCADAPack 575	10 Counter Inputs (Shared with Dis)	01...02 controller board lower left 03...10 I/O board lower left

7.2.4 Analog Inputs

Unit	I/O	Terminal Location
SCADAPack 470	4 Analog Inputs	01...04 controller board upper left
SCADAPack 474	12 Analog Inputs	01...04 controller board upper left 05...12 I/O board lower left
SCADAPack 570	---	---
SCADAPack 574	8 Analog Inputs	01...08 I/O board bottom center
SCADAPack 575	6 Analog Inputs	01...06 I/O board bottom center

7.2.5 Analog Outputs

Unit	I/O	Terminal Location
SCADAPack 470	---	---
SCADAPack 474	2 Analog Outputs	01...02 I/O board lower left
SCADAPack 570	---	---
SCADAPack 574	2 Analog Outputs	01...02 I/O board lower left
SCADAPack 575	2 Analog Outputs	01...02 I/O board lower right

7.3 Communication Ports

See:

- Serial Communications Ports 20
- Ethernet Communications Ports 21
- USB Communication Ports 21

7.3.1 Serial Communications Ports

Unit	Ports	Port Location

SCADAPack 470	Ports 1, 2 RS-485, 2 W Ports 3, 4 RS-232 or RS-485 2 W Port 5 RS-232	All Controller board lower right. Ports 1, 2 use terminal blocks, 3, 4 are RJ45 Port 5 internal
SCADAPack 474	Ports 1, 2 RS-485 2 W Ports 3, 4 RS-232 or RS-485 2 W Port 5 RS-232	All Controller board lower right. Ports 1, 2 use terminal blocks, 3, 4 are RJ45 Port 5 internal
SCADAPack 570	Ports 1, 2 RS-232 Ports 3, 4 config as RS-232 / RS485 2 W	Controller board lower right. RS-232 are RJ45, RS-485 are terminals
SCADAPack 574	Ports 1, 2 RS-232 Ports 3, 4 config as RS-232 / RS485 2 W	Controller board lower right. RS-232 are RJ45, RS-485 are terminals
SCADAPack 575	Ports 1, 2 RS-232 Ports 3, 4 config as RS-232 / RS485 2 W	Controller board lower right. RS-232 are RJ45, RS-485 are terminals

7.3.2 Ethernet Communications Ports

Unit	Ports	Port Location
SCADAPack 470	LAN1, LAN2	Controller board upper right
SCADAPack 474	LAN1, LAN2	Controller board upper right
SCADAPack 570	LAN1, LAN2, LAN3	Controller board upper right
SCADAPack 574	LAN1, LAN2, LAN3	Controller board upper right
SCADAPack 575	LAN1, LAN2, LAN3	Controller board upper right

7.3.3 USB Communication Ports

Unit	Ports	Port Location
SCADAPack 470	USB-C Device port for programming	Controller board upper center Controller board upper center

	USB-A Host port for storage up to 32 GB	
SCADAPack 474	USB-C Device port for programming USB-A Host port for storage up to 32 GB	Controller board upper center Controller board upper center
SCADAPack 570	USB-B Device port for programming USB-A Host port for storage up to 32 GB	Controller board upper center Controller board upper center
SCADAPack 574	USB-B Device port for programming USB-A Host port for storage up to 32 GB	Controller board upper center Controller board upper center
SCADAPack 575	USB-B Device port for programming USB-A Host port for storage up to 32 GB	Controller board upper center Controller board upper center

7.4 Power Supply

Unit	Requirement	Terminal Location
SCADAPack 470	14…29 Vdc 2.8 W	Controller board bottom left
SCADAPack 474	14…29 Vdc Max 8.4 W	Controller board bottom left and I/O board bottom left
SCADAPack 570	12…30 Vdc typ 4.3 W	Controller board bottom left
SCADAPack 574	12…30 Vdc typ 6.5W Max 9.2 W	Controller board bottom left and I/O board bottom right
SCADAPack 575	12…30 Vdc typ 5.4W Max 9.1 W	Controller board bottom left and I/O board bottom right

8 ISaGRAF 3.55 to Workbench Target 5 Considerations

ISaGRAF Target 3 applications are not compatible with SCADAPack Workbench Target 5 applications. There is no automated mechanism to convert an ISaGRAF application into a Workbench Target 5 application. The purpose of this guide is to facilitate the conversion to the Workbench Target 5 format, such that it may then be uploaded to the EcoStruxure Control Engineering Converter website for automated conversion to a SCADAPack x70-compatible project. The project produced by the EcoStruxure Control Engineering Converter website can then be used to expedite the full conversion of the project originally created using ISaGRAF 3.

The ISaGRAF communication and I/O configuration cannot be directly transferred into a SCADAPack RemoteConnect application. It may be easiest to re-create these settings in the RemoteConnect application. If, however, a SCADAPack E Configurator *.rtu file is made available, it may be uploaded into the EcoStruxure Control Engineering Converter along with the Workbench Target 5 application. This will populate the RemoteConnect application's DNP3 points and Modbus registers, if any.

See:

- ISaGRAF 3.55 23
- ISaGRAF Versions Prior to ISaGRAF v3.55 23
- SCADAPack Workbench Target 5 23
- SCADAPack ISaGRAF and Workbench Differences 23

8.1 ISaGRAF 3.55

ISaGRAF (Target 3) applications are generated using ISaGRAF 3.55. This is a 16-bit application which runs best on Windows XP or Windows 7 32-bit. For newer computers it may be best to run it in a Windows 7 Virtual Machine.

8.2 ISaGRAF Versions Prior to ISaGRAF v3.55

For ISaGRAF projects prior to ISaGRAF v3.55, it is recommended to use ISaGRAF Workbench v3.55 to import and upgrade projects to that version first; ISaGRAF 3.55 is the starting point for this guide.

Earlier versions of ISaGRAF projects may also be progressed to ISaGRAF Target 5 projects using this guide, however intricacies and incompatibilities present in versions of ISaGRAF prior to v3.55 have not been considered.

8.3 SCADAPack Workbench Target 5

There are significant differences between the two targets, in both the built-in ISaGRAF functions and the SCADAPack E-Series product-specific functions provided by Schneider Electric.

8.4 SCADAPack ISaGRAF and Workbench Differences

There are a number of differences between ISaGRAF and SCADAPack Workbench Target 5. These differences need to be taken into account when manually porting an ISaGRAF application to SCADAPack Workbench Target 5.

9 Converting a Target 3 Application to a Target 5 Application

This section provides details on how to convert an ISaGRAF Target 3 application to a SCADAPack Workbench Target 5 application. It should be noted that this guide outlines the main steps to import the application variables and source code, however all source code should be reviewed against the differences detailed in SCADAPack Workbench Differences 61.

See:

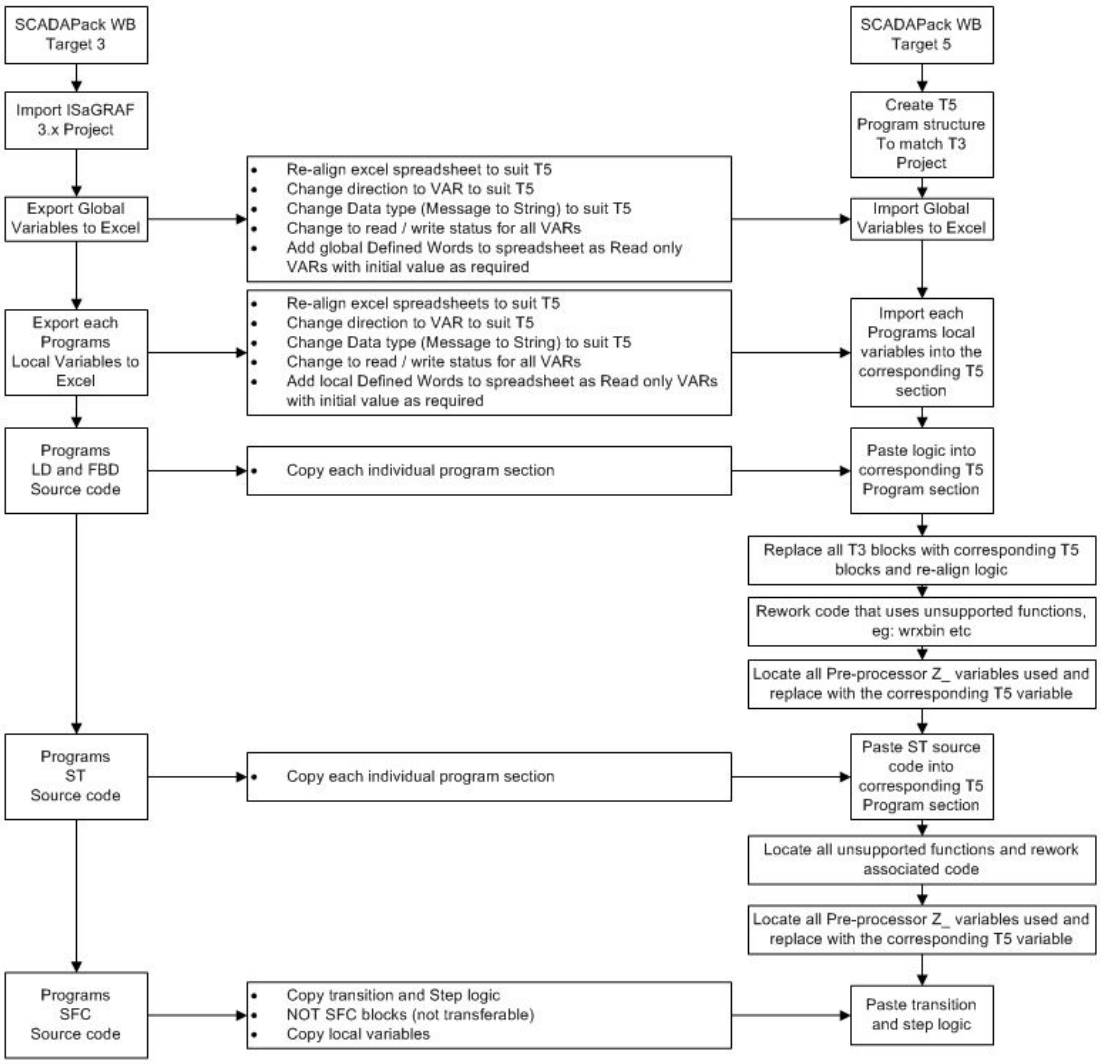
- Outline of the Porting Method 24
- SCADAPack Configurator Versions 35

9.1 Outline of the Porting Method

The method used to port an ISaGRAF 3.55 application to a Workbench Target 5 application can be considered in three main steps:

1. Import the ISaGRAF application into SCADAPack Workbench as a Target 3 project.
2. Use a combination of manual operations to port the variables and logic sections into a Workbench Target 5 application.
3. Upload the Target 5 application to Schneider Electric's EcoStruxure Control Engineering Converter tool to generate a SCADAPack RemoteConnect application.
4. The flowchart below outlines the basic method for porting ISaGRAF 3.55 User Function Blocks to Workbench Target User Function Blocks. This document focuses on this process.

NOTE: ISaGRAF applications imported into SCADAPack Workbench cannot be downloaded to a SCADAPack E RTU, but this is only an intermediate step. The application created will only be used to port the ISaGRAF application to the Target 5 application.



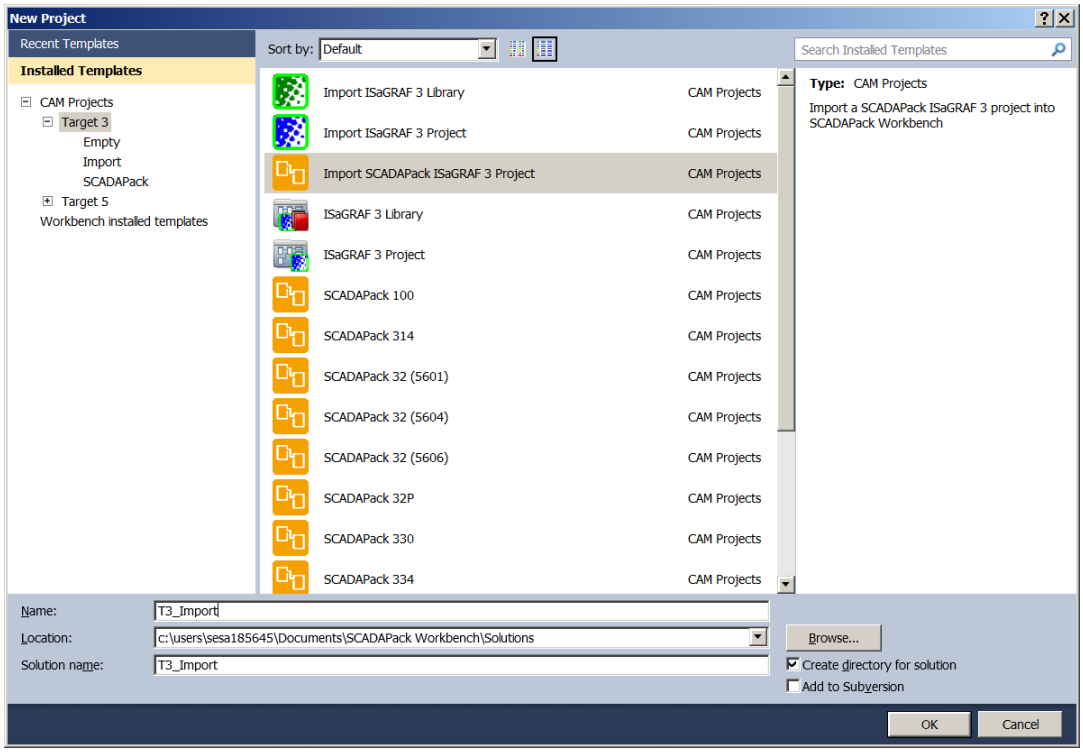
See:

- Importing ISaGRAF 3.55 Project 26
- Setting up the Workbench Target 5 Project 27
- Creating I/O Devices 29
- Transferring Global Variables 29
- Transferring Program Local Variables 32
- Transferring FBD and LD Program Source Code 33
- Transferring ST Program Source Code 34

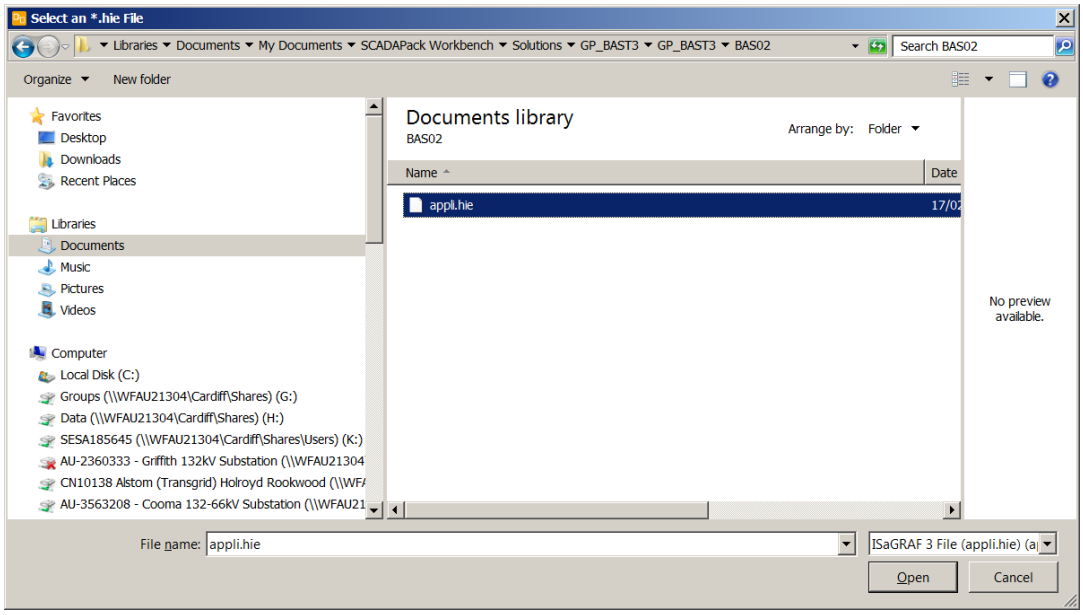
9.1.1 Importing ISaGRAF 3.55 Project

To import the ISaGRAF 3.55 project

1. Open SCADAPack Workbench and select **File > New > Project**.



2. Select Installed Template **Target 3** and **Import SCADAPack ISaGRAF 3 Project**.
3. Enter a name for the project, then Browse to the folder where you want to save it.
4. Click **OK**.
5. Navigate to the folder containing the ISaGRAF 3 project to be imported, and select the **appli.hie** file.



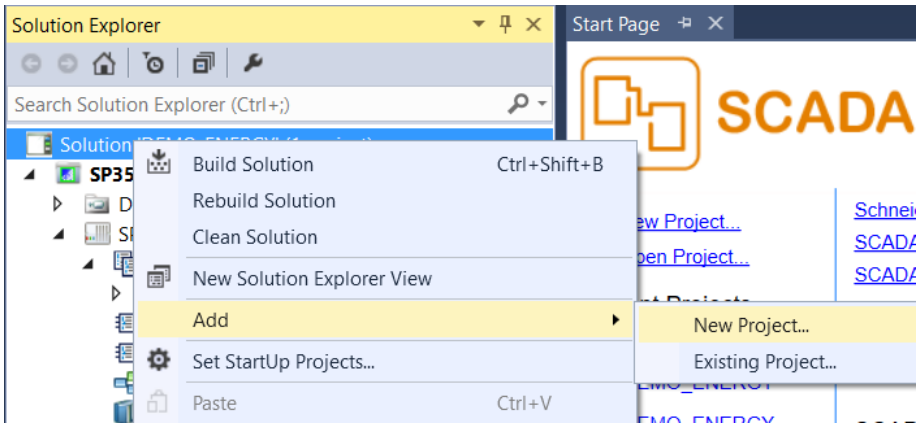
NOTE: You can find more details on importing ISaGRAF Target 3 projects in the **SCADAPack Workbench Manuals**, within the **Migrating ISaGRAF 3 Workbench Projects to SCADAPack Workbench** section.

9.1.2 Setting up the Workbench Target 5 Project

To create a new Target 5 Project in SCADAPack Workbench

1. At the top of the Solution Explorer window, just above the imported ISaGRAF project, right-click **Solution**.
2. Select **Add > New Project**.

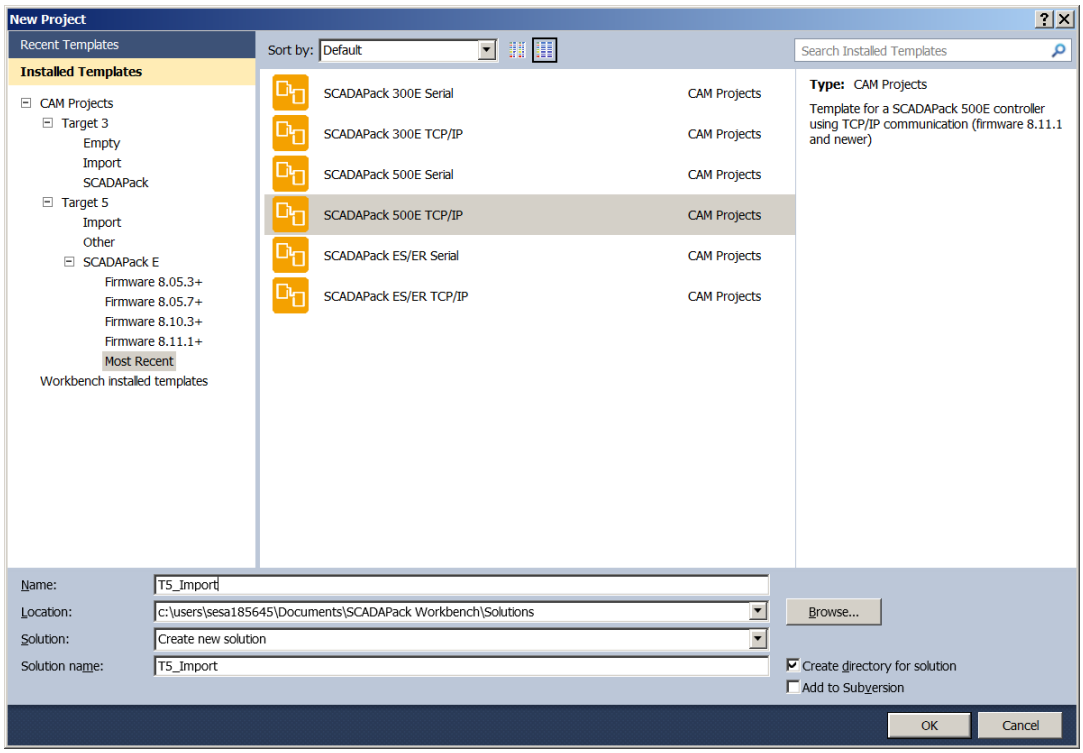
   NOTE: Keep the imported Target 3 project open so that both projects are available at the same time.



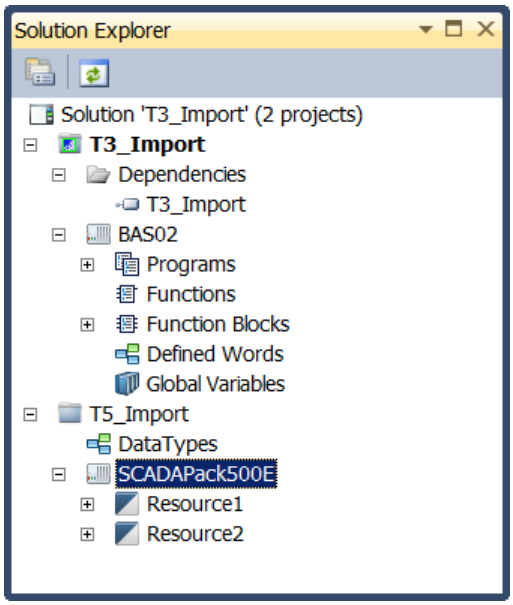
3. In the **New Project** dialog, expand the **Target 5** category.

4. Within the SCADAPack E list, select the **Most Recent** firmware version and the desired RTU type.

5. Enter a project name, browse to the folder where you want the project to be saved and click **Select Folder**.



The **Solution Explorer** window now shows the imported ISaGRAF project and the new Target 5 project that has been created.



6. Create a program structure in the Target 5 project which matches the structure of the Target 3 project as closely as possible.

   Following steps transfer the contents of the Global and Local program variables into the Target 5 project, and copy the contents of each program.

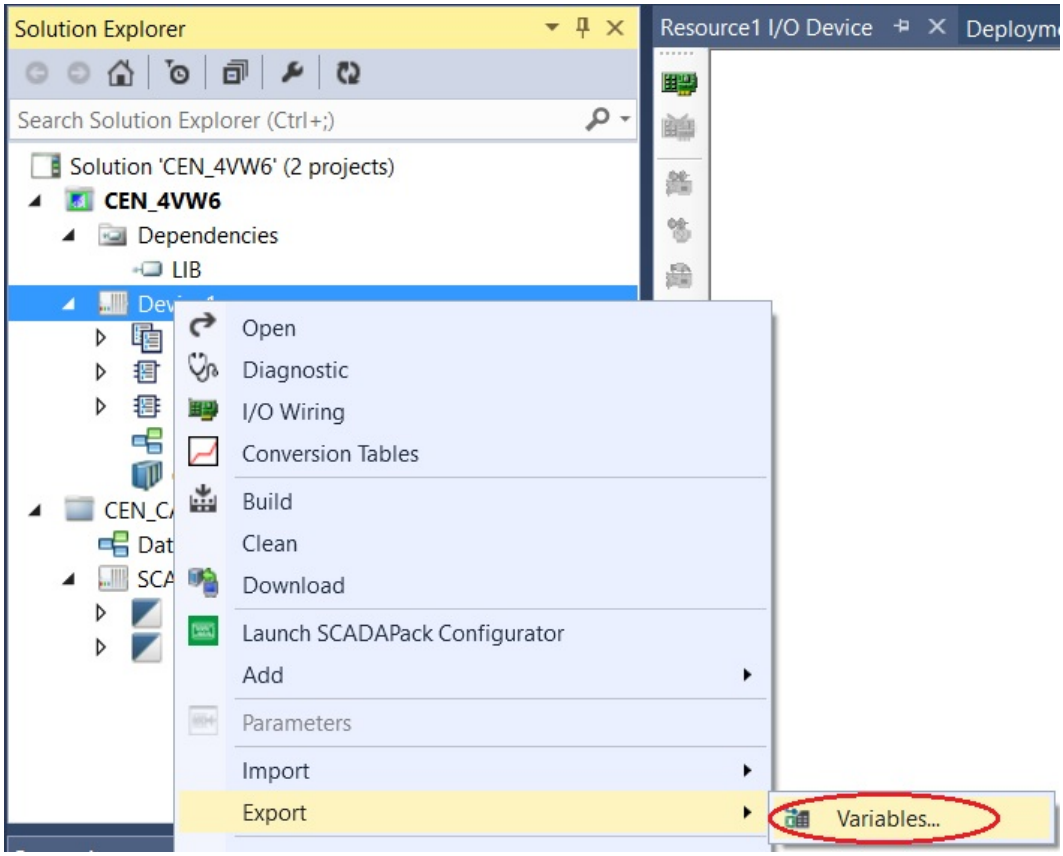
9.1.3 Creating I/O Devices

1. In the newly-created Target 5 application, create the required I/O devices to match the ISaGRAF 3.55 I/O board mappings.
2. In Workbench Target 3, access the existing hardware (I/O board) by right-clicking on the RTU **Device** and selecting the **I/O Wiring** option.
3. In Target 5, access the existing hardware (I/O board) found by right-clicking on the project's **Resource1** entry (or **Resource2** if that were used) and selecting **I/O Device**.
4. Refer to the **SCADAPack E Target 5 Help** documentation for further information.
5. Navigate to **Target 5 I/O Device Technical Reference Manual > I/O Devices**. The sub-section RTU Database I/O Devices provides information about reading from and writing to digital, analog, and counter points.

NOTE: As Workbench Target 5 can have a variety of I/O device sizes (up to 100 I/O), a number of Target 3 boards can be consolidated into a single Target 5 I/O Device. You may want to consolidate the I/O boards at this stage for easier management of the project.

9.1.4 Transferring Global Variables

1. Export the Target 3 **Global Variables** by right-clicking the Target 3 Device and selecting **Export > Variables.**



2. Browse to the desired location to save the file, enter a filename for the exported Excel spreadsheet, then click **Save**.

3. Open the **Global Variables** spreadsheet.

   It contains the following column headings:

   - Name
   - DataType
   - StringSize
   - Dimension
   - Wiring
   - Attribute
   - Direction
   - ModbusAddress
   - IsRetained
   - InitialValue
   - Unit

- Comment

4. Create a new spreadsheet for the import to Workbench Target 5.
   a. Export from the empty Target 5 project.
   b. Right-click **Resource1**.
   c. Select **Export > Variables**.
5. Copy the contents, with the exception of the wiring data, from the Workbench Target 3 spreadsheet to the corresponding columns of the Target 5 spreadsheet.

   This new spreadsheet contains the following columns:

   NOTE: Be careful to paste into the correct column, as the columns are not in the same sequence in the two spreadsheets.

o Name	Data copied from T3 column ‘Name’
o DataType	Data copied from T3 column ‘DataType’
o Dimension	Data copied from T3 column ‘Dimension’
o StringSize	Data copied from T3 column ‘StringSize’
o InitialValue	Data copied from T3 column ‘InitialValue’
o Direction	Data copied from T3 column ‘Direction’
o Attribute	Data copied from T3 column ‘Attribute’
o Comment	Data copied from T3 column ‘Comment’
o Alias	No Data copied
o Wiring	Do NOT copy data for this column
o Address	No Data copied
o Retained	Data copied from T3 column ‘IsRetained’
o RetainedFlags	No Data copied
o Groups	No Data copied

6. Perform the following changes to the Workbench Target 5 spreadsheet:
   a. Set Direction to **Var** for all variables with the Attribute **ReadWrite**
   b. Set Direction to **VarInput** for all variables with the Attribute **Read**
   c. Set Direction to **VarOutput** for all variables with the Attribute **Write**
   d. Change TextDataType **Message** to **String**

NOTE: Any Global Defined Words cannot be exported in SCADAPack Workbench. These need to be copied manually, if necessary.

7. Save the Workbench Target 5 Global Variables spreadsheet. You can then import it into the Target 5 Global variables.
8. Right-click the Target 5 **Resource1** folder and select **Import > Variables**.
9. Navigate to the spreadsheet just created, select it, then click **Open**.
10. Check the Output window for any warning or error messages. The last line reads (for example) “Import: xxxx succeeded, xx warnings, x failed”. If required, make any necessary corrections or repairs.

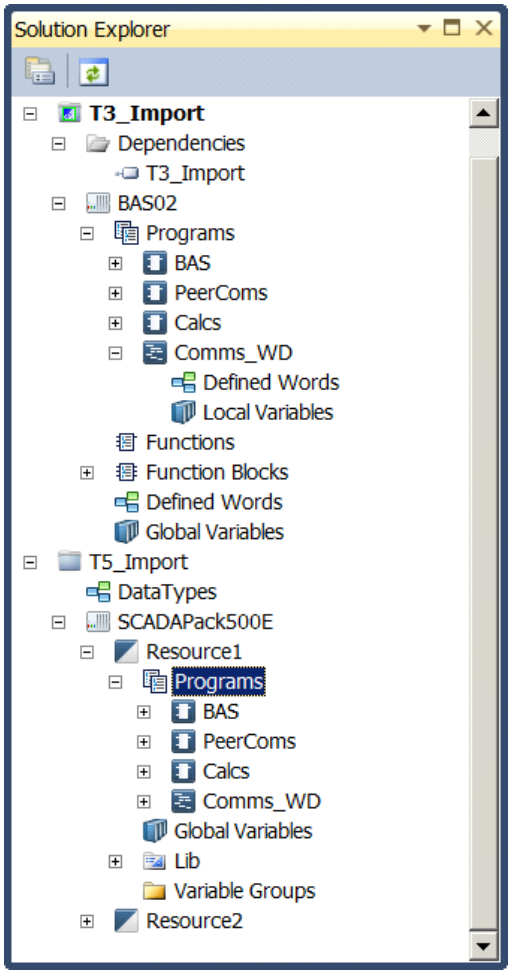
    For example, arrays may not be properly imported. Also, special features only available in Target 3 (e.g., DLOG for the Standard SCADAPack) do not import properly. These need to be re-built once the application conversion process has been completed, within SCADAPack RemoteConnect. See Target 3 Functions and Function Blocks Not Supported in Target 5 66.

9.1.5 Transferring Program Local Variables

Before importing the Local Variables for each program, create a program structure in the Workbench Target 5 project to match the Target 3 project as closely as possible.

The method for exporting from the Target 3 application and importing into Target 5 is the same as for the Global Variables, with the exception that a separate export / import needs to be performed for each program. Right-click on each program, select **Export > Variables**.

The Local Defined Words are added to the Workbench Target 5 spreadsheet using the same method as with Global Defined Words.



9.1.6 Transferring FBD and LD Program Source Code

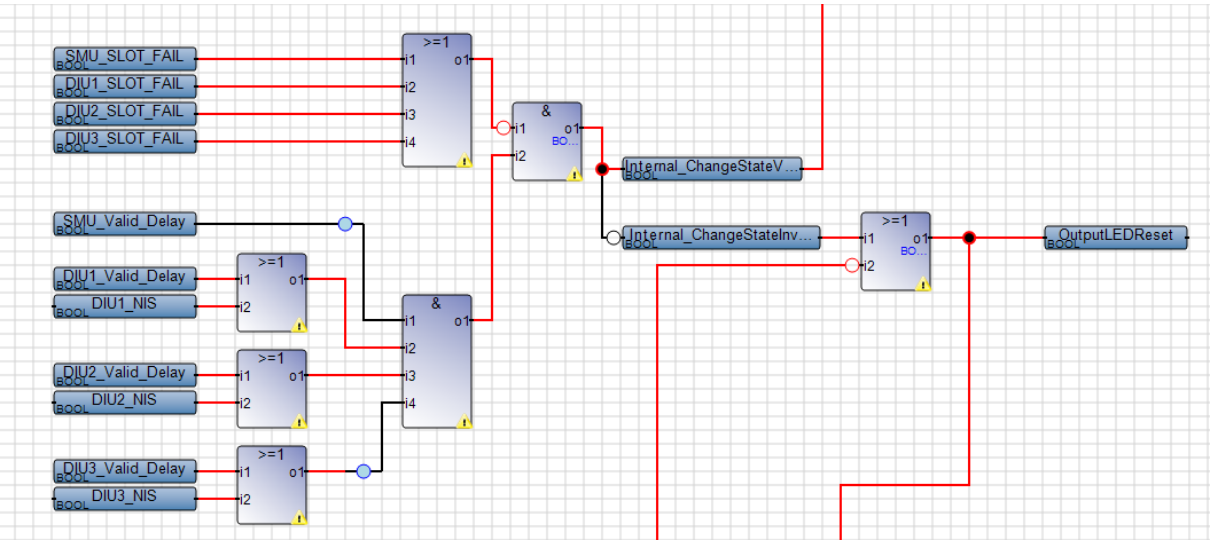
1. For each Target 3 program, double-click on it to open the editor.
2. Copy the contents of the editor:
   a. Select All [Ctrl-A]
   b. Copy [Ctrl-C]
3. Open the corresponding Workbench Target 5 program editor and paste the contents copied from the T3 program:
   a. Paste [Ctrl-V]
4. Close and save the Workbench Target 5 program and then reopen it.

   This is necessary when changing function blocks to avoid the blocks from becoming disconnected in the editor.

   As can be seen from the code snippet below, the code still contains Target 3 function blocks. These blocks need to be exchanged for the equivalent Workbench Target 5 blocks.

For example, replace '>=1' blocks with 'OR' and '&' blocks with 'AND'.

The mismatched blocks (and any blocks exhibiting an error) are highlighted with a yellow exclamation mark warning triangle.



The next step is to locate all unsupported blocks and to modify the code to replace the unsupported blocks with code using supported blocks Again, these blocks are indicated with the yellow exclamation mark warning triangle. See SCADAPack Workbench Differences 61 for more details on this process.

9.1.7 Transferring ST Program Source Code

1. For each program, double-click on the Workbench Target 3 program folder to open the editor.
2. Copy the contents of the editor:
   a. Select All [Ctrl-A]
   b. Copy [Ctrl-C]
3. Open the corresponding Target 5 program editor and paste the contents copied from the Target 3 program:
   a. Paste [Ctrl-V]
4. Locate all unsupported functions and modify the code to replace the unsupported functions with code using supported functions.

   Unsupported functions appear in the local variables list with a DataType of (Undefined).

For example, modify code using TSTART and TSTOP functions to use TON function block. See SCADAPack Workbench Differences 61 for more details. Repeat the procedure for each ST program.

9.2 SCADAPack Configurator Versions

Certain configuration parameters cannot be configured within either ISaGRAF 3.55 or Workbench. These include communication port settings, protocol settings, and miscellaneous other items. For each version of software and RTU firmware, this need to be managed differently.

While following the porting instructions for the application, these configuration items may also be transferred by copying them manually to SCADAPack E Configurator configured for Target 5 mode.

ISaGRAF 3.55 for Standard SCADAPacks

These items are configured within a custom sub-menu within ISaGRAF itself, under **Tools > Controller**.

Workbench for Standard SCADAPacks

These items are configured with an associated software package, created by Schneider Electric, called SCADAPack Configurator.

ISaGRAF 3.55 for E Series SCADAPacks

These items are configured with the SCADAPack E Configurator, a different but related application. This software needs to be set for Target 3 mode.

Workbench for E Series SCADAPacks

These items are also configured with SCADAPack E Configurator. However, the software needs to be set for Target 5 mode.

10 Converting Target 3 User Function Blocks to Target 5 FBs

This section explains how to port ISaGRAF Target 3 User Function Blocks to Workbench Target 5 User Function Blocks. It should be noted that this guide outlines the basic steps to import the application variables and source code and that all source code needs to be reviewed against the differences detailed in SCADAPack Workbench Differences 61.

See:

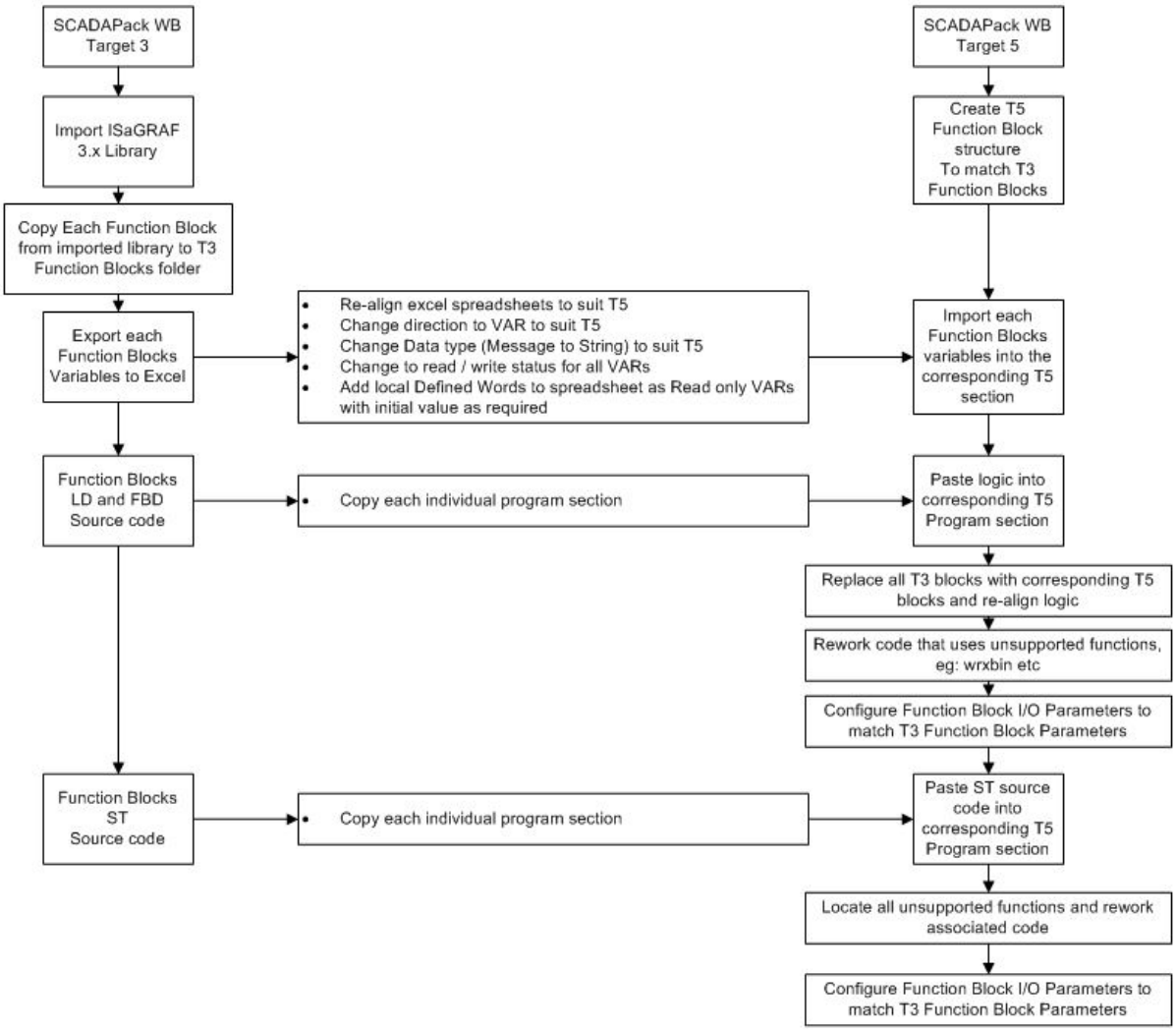
- Outline of the Porting Method 36
- Detailed Process Steps 37

10.1 Outline of the Porting Method

The method used to port a Target 3 User Function Block to a Target 5 User Function Block can be considered in two main steps:

1. Import the library as an ISaGRAF Target 3 library into SCADAPack Workbench.
2. Via a manual process, using a combination of methods to port the Function Blocks, complete with variables and code.

The flowchart below outlines the basic method for porting ISaGRAF Target 3 user-defined Function Blocks to Workbench Target 5 user-defined Function Blocks.



10.2 Detailed Process Steps

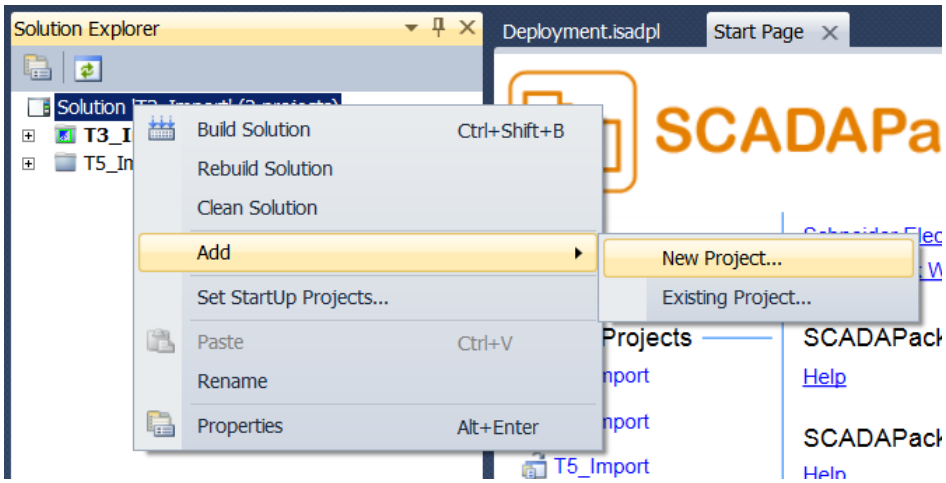
See:

- Importing Target 3 Library 37
- Creating a Target 5 Library 40
- Transferring Function Block Variables 42
- Transferring FBD and LD Program Source Code 43
- Transferring ST Program Source Code 43
- Configuring I/O Parameters 44

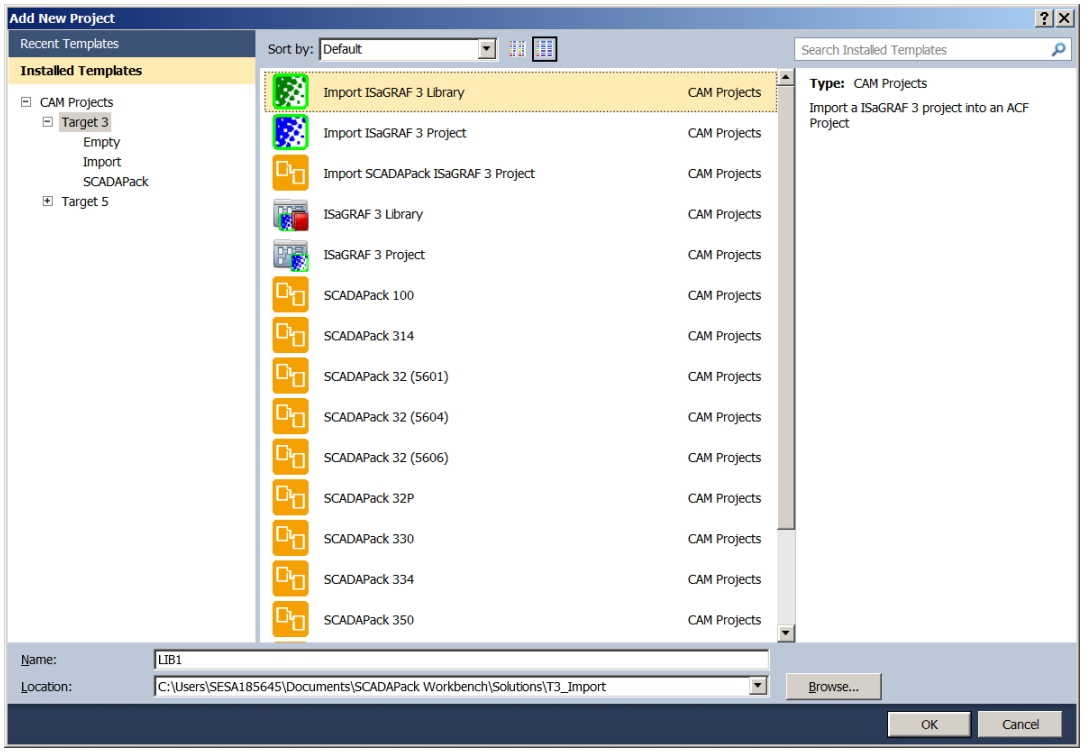
10.2.1 Importing Target 3 Library

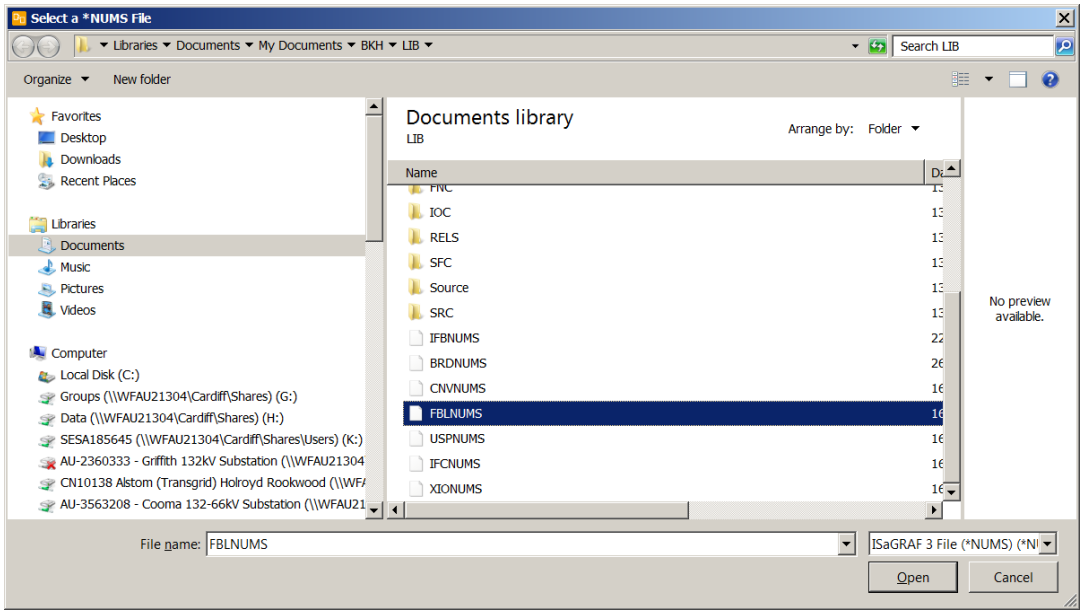
1. Copy the ISaGRAF Target 3 LIB folder (usually located in C:\ISAWIN\LIB) to a convenient location.

2. In SCADAPack Workbench right-click on the **Solution** level and select **Add > New Project**.

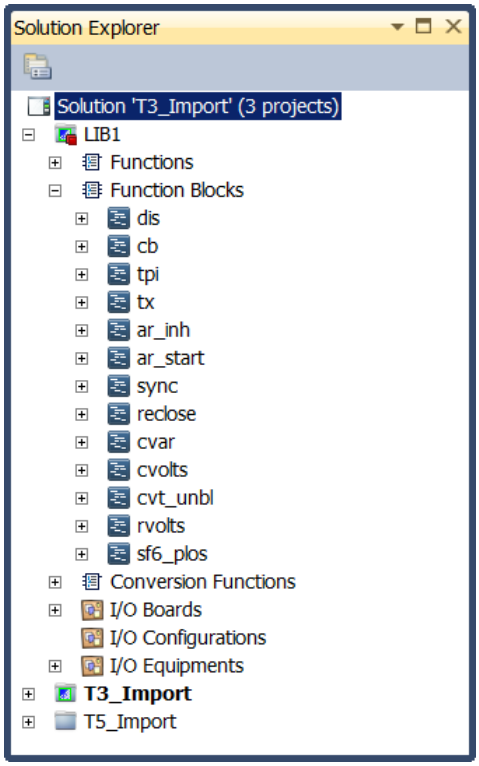


3. In the **Add New Project** window select **Target 3** and **Import ISaGRAF 3 Library**.





4. Select the location of the ISaGRAF 3 Library and then one of the NUMS files and click **Open**.

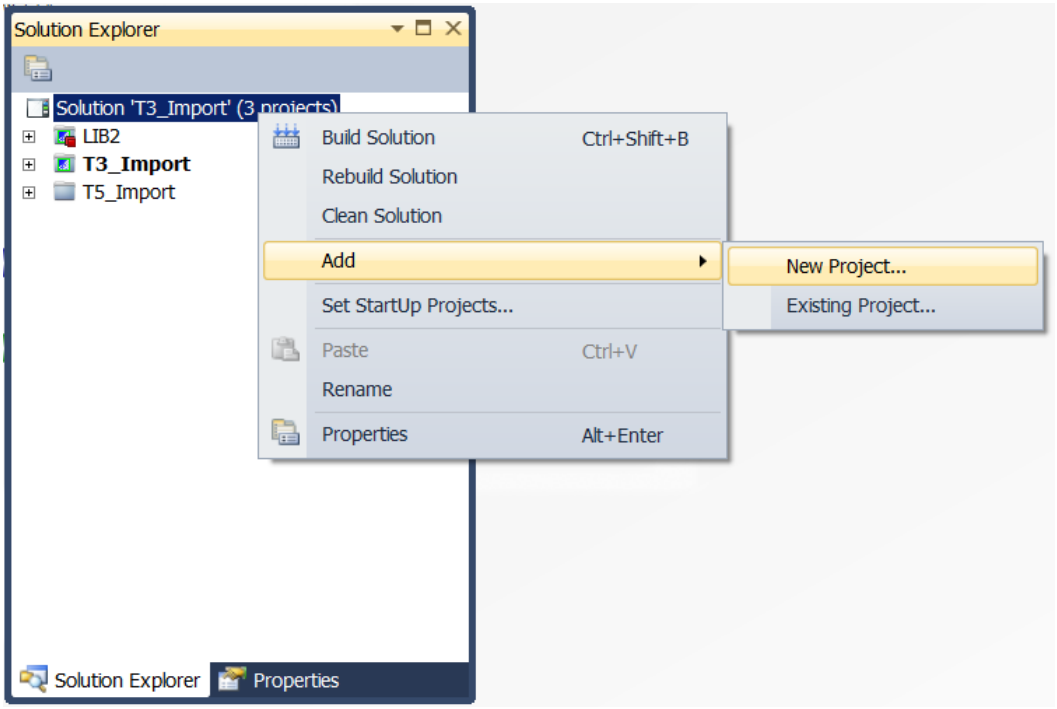


This process can take a few minutes to complete. When complete, you can see the library in the Solution window.

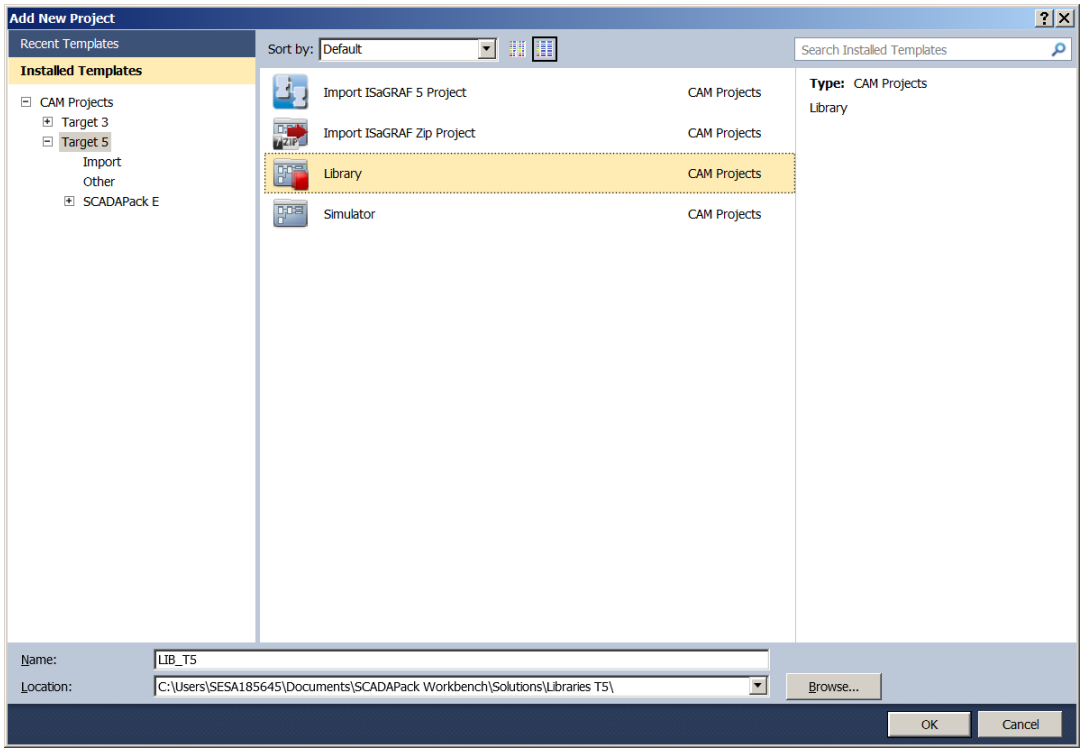
The user-defined Function Blocks are found in the folder **LIB1/Function Blocks**.

10.2.2 Creating a Target 5 Library

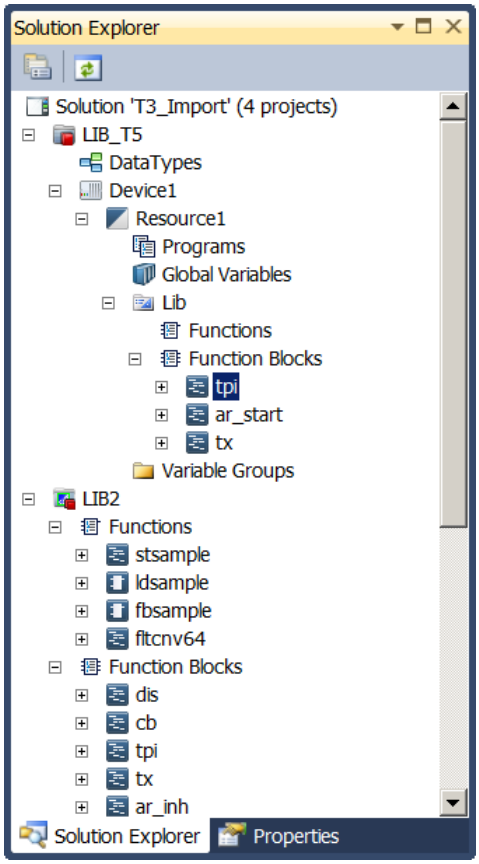
1. Create the Target 5 Library by right-clicking on the solution and selecting **Add > New Project**.



2. Select template **Target 5 – Library**.
3. Select a folder separate from the project folder where the libraries will reside and a library name.



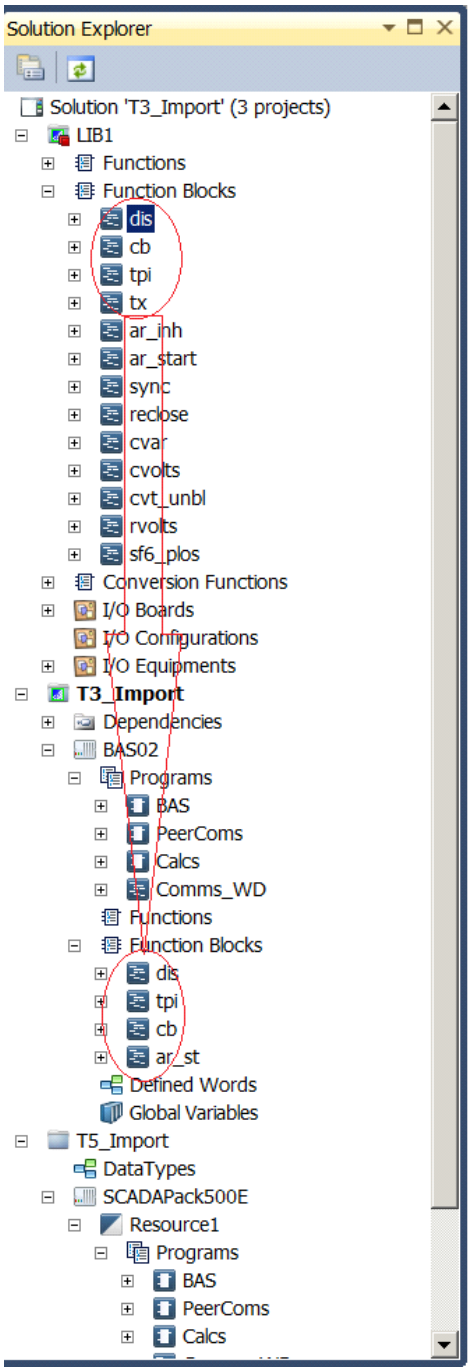
The following shows an example function block library structure for Target 3 and Target 5.



4. In the newly created library create the function blocks required. The code type (e.g., ST, FBD, etc.) needs to match that of the corresponding Target 3 function blocks.

10.2.3 Transferring Function Block Variables

Due to a known issue in SCADAPack Workbench, the Function Block Variables cannot be exported from the LIB folder. To work around this, copy the required function blocks from the LIB folder to the 'Function Blocks' folder in the Target 3 Project.



1. Right-click on the Function Block to be imported and export the variables.
2. Follow the method outlined in Transferring Global Variables 29 to import the variables for each required User Function Block into the newly created Target 5 Library.

10.2.4 Transferring FBD and LD Program Source Code

1. For each user-defined function block double-click on the function block in the Target 3 library to open the editor.
2. Copy the contents of the editor:
   a. Select All [Ctrl-A]
   b. Copy [Ctrl-C]
3. Open the editor for the corresponding user-defined function block in the Workbench Target 5 library and past the contents copied from the Target 3 function block.
   a. Paste [Ctrl-V]
4. Close and save the Target 5 function block and then reopen it.

   This is necessary when changing function blocks, to avoid the blocks from becoming disconnected in the editor.
5. Replace all the Target 3 blocks with the equivalent Workbench Target 5 blocks.

   For example, replace '>=1' blocks with 'OR' and '&' blocks with 'AND'. The mismatched blocks are highlighted with a yellow triangle.
6. Locate all unsupported blocks and modify the code to replace the unsupported blocks with code using supported blocks.

   See Supported Functions and Function Blocks 71 for a list of supported and unsupported blocks.
7. Repeat the procedure for each FBD and LD program folder.

10.2.5 Transferring ST Program Source Code

1. For each user-defined function block double-click on the Target 3 function block folder to open the editor.
2. Copy the contents of the editor:
   a. Select All [Ctrl-A]
   b. Copy [Ctrl-C]
3. Open the corresponding Target 5 function block editor and past the contents copied from the T3 program.
   a. Paste [Ctrl-V]

4. Locate all unsupported functions and to modify the code to replace the unsupported functions with code using supported functions.

   Unsupported functions appear in the local variables list with a DataType of '(Undefined)'

5. Modify code using TSTART and TSTOP functions to use TON function block.

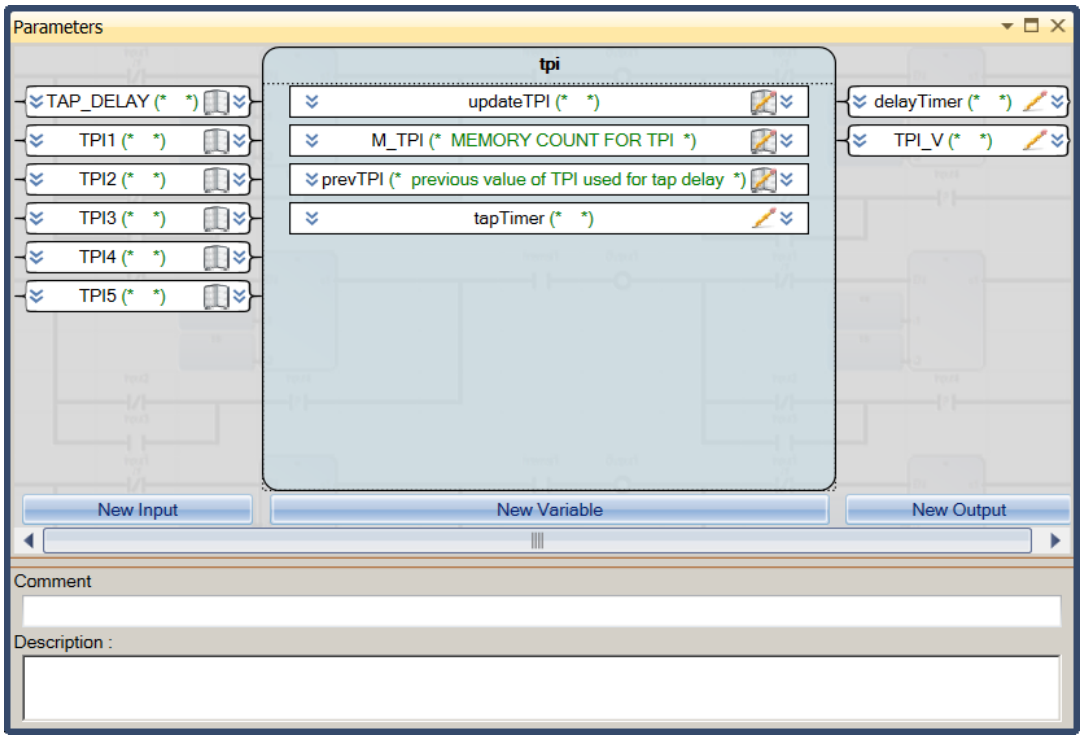
   See SCADAPack Workbench Differences 61 a list of supported and unsupported blocks.

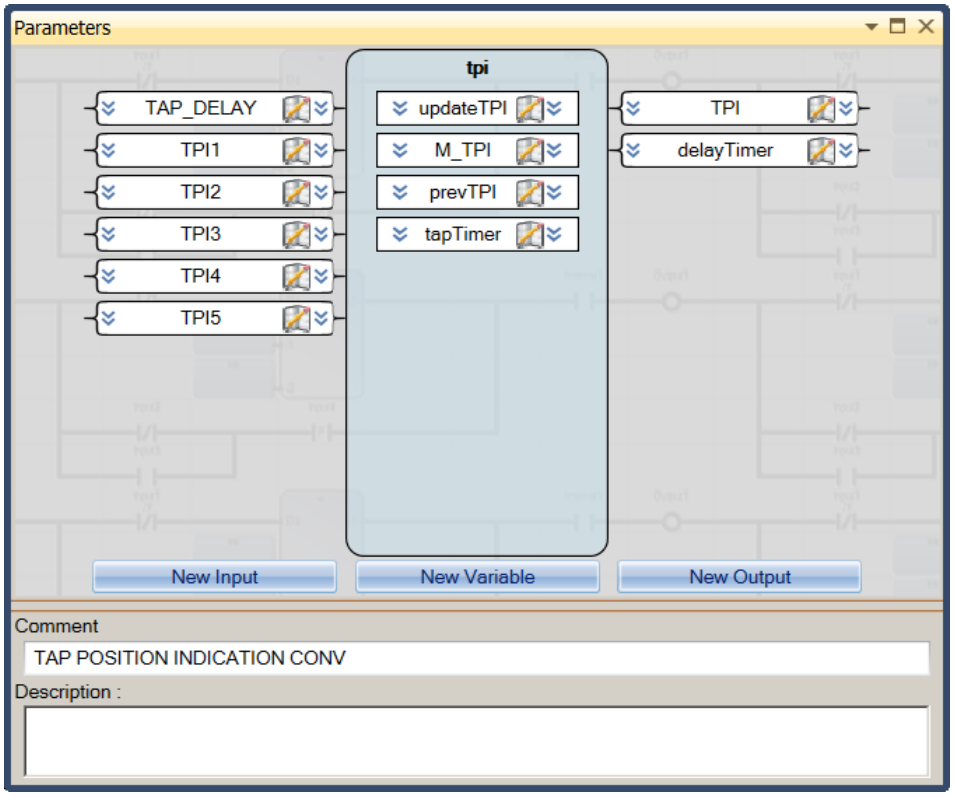
6. Repeat the procedure for each ST User Defined Function Block.

10.2.6 Configuring I/O Parameters

1. Right-click on the function block and select parameters.
2. Check that the input and output parameters are configured the same as those of the Target 3 Function Block.

   The following screen shots show the Target 5 and Target 3 I/O parameter windows.




Parameters
tpi
TAP_DELAY
TPI1
TPI2
TPI3
TPI4
TPI5
updateTPI
M_TPI
prevTPI
tapTimer
TPI
delayTimer
New Input
New Variable
New Output
Comment
TAP POSITION INDICATION CONV
Description :

11 Converting Workbench Application to SCADAPack RemoteConnect

See:

- Using the EcoStruxure Control Engineering Website 46
- Importing the Converted Application 56
- Verifying Application Functionality 59

11.1 Using the EcoStruxure Control Engineering Website

Once the Workbench Target 5 application has been created, and modified as required, the next step is to upload the application to Schneider Electric's EcoStruxure Control Engineering Converter website. The Converter is an automatic tool for translating an application to a different controller brand, model or language, while maintaining the same operation as the source application.

For technical assistance with the EcoStruxure Control Engineering website, see Technical Support 7.

See:

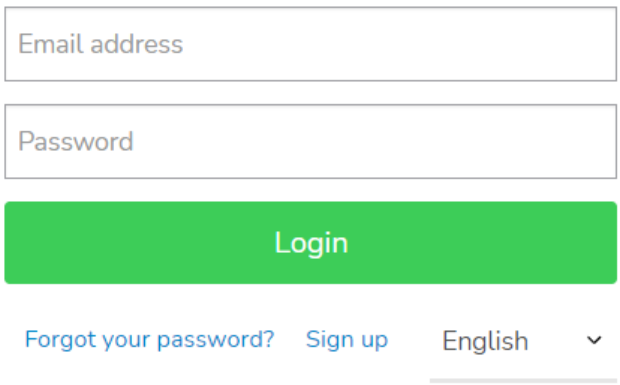
- Registering on the EcoStruxure Control Engineering Website 46
- Purchasing a Conversion License Activation Key 47
- Performing a Typical Application Conversion 48
- Converting a Project Containing Library Dependencies 54

11.1.1 Registering on the EcoStruxure Control Engineering Website

Before accessing the tool, you need to create an account on the website.

1. Open a web browser and navigate to this address: https://ecostruxure-control-engineering.se.app/users/login
2. Click the **Sign up** link, as shown below.
3. Complete the registration form and click **I agree to the Terms & Conditions**.
4. Click the green **Sign Up** button at the bottom of the dialog.

   You will receive an Account Registration confirmation email shortly.
5. It typically takes about one business day for the Account Activation confirmation email to be received.



11.1.2 Purchasing a Conversion License Activation Key

Using the conversion service requires purchase of a license activation key for each conversion that will be performed. Each separate Workbench Target 5 application that will be converted requires its own license activation key. However, once an application has been converted it may be used in as many controllers as you require.

If the existing RTU is running two targets (applications), you will need two licenses.

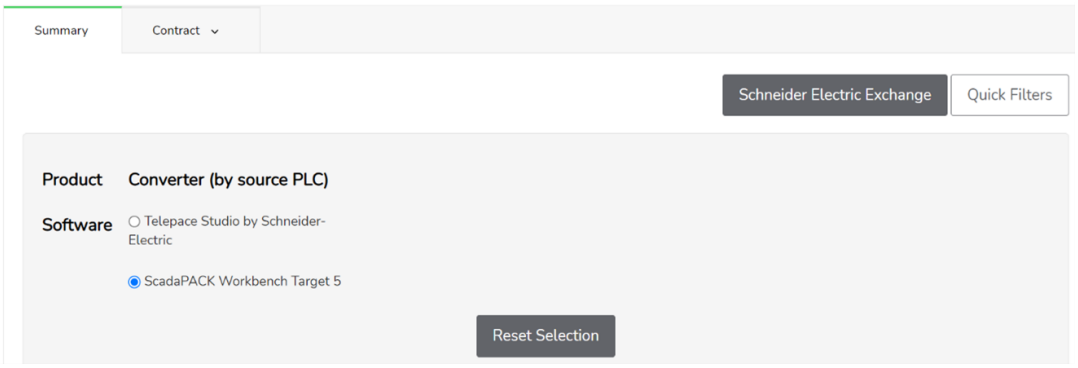
Purchase the license activation key through the customer's normal sales channel, using the part number:

TBUM297154

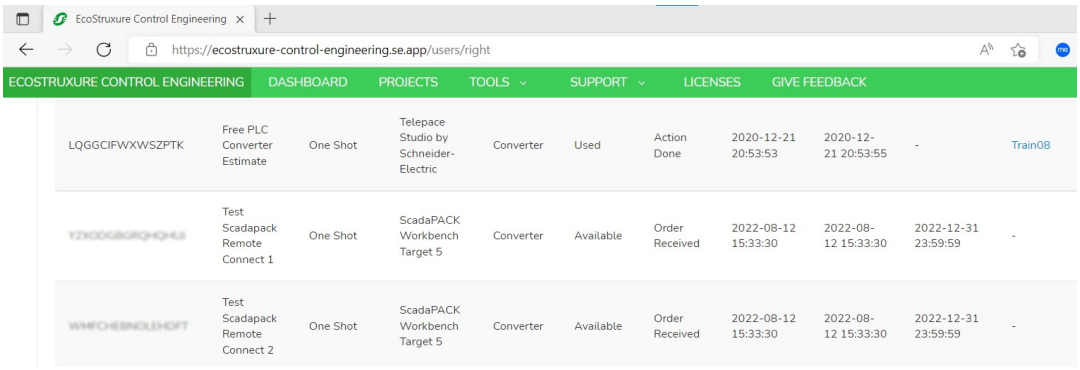
When placing an order for a license activation key, you need to include the same email address (username) that was provided when first registering on the EcoStruxure Control Engineering website

Once the license purchase confirmation email has been received, the license will be available on the EcoStruxure Control Engineering Converter website. After logging in to the website, the first page displayed is the Dashboard. It provides an overview of the user's activity.

1. To view the available licenses, click on the **LICENSES** menu item at the top of the page.
2. To see only the licenses for SCADAPack Workbench, click **Quick Filters**.

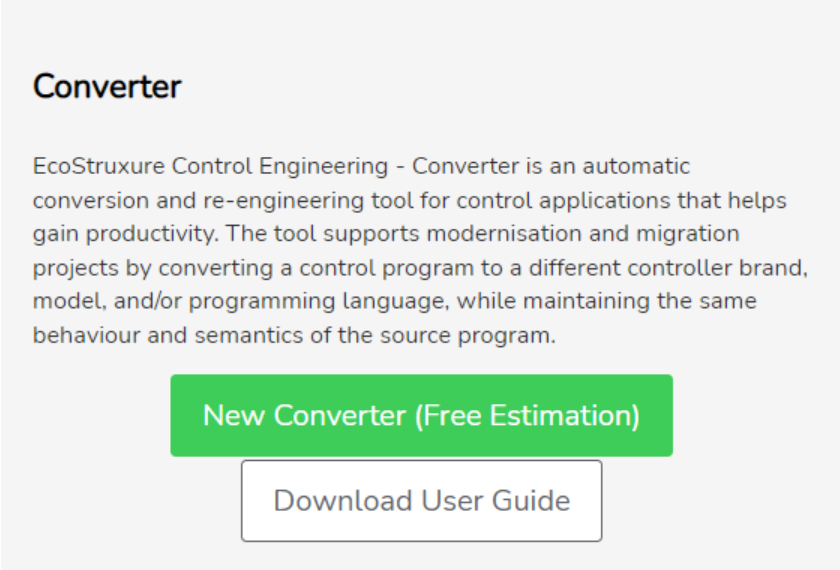


3. Scroll down to view the list of available licenses.

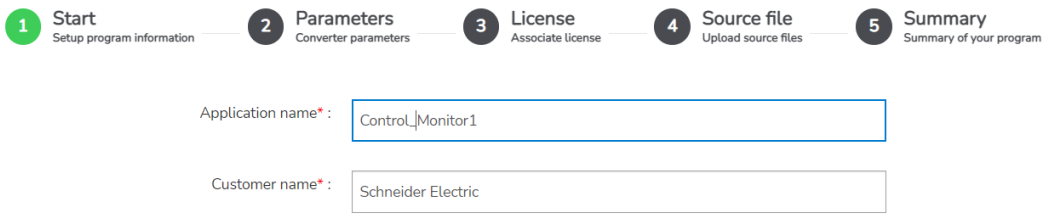


11.1.3 Performing a Typical Application Conversion

1. Scroll down to the Converter area on the Dashboard page.
2. Click the green **New Converter (Free Estimation)** button.

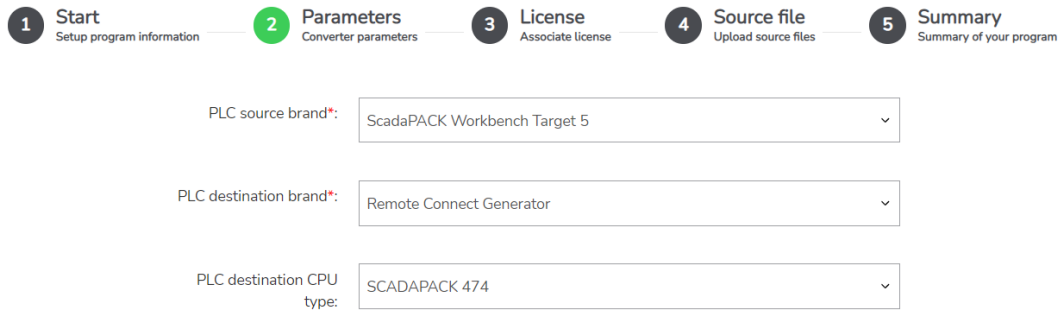


3. Enter the Application **name** and **Customer name**.
4. Click **Validate and Next Step**.



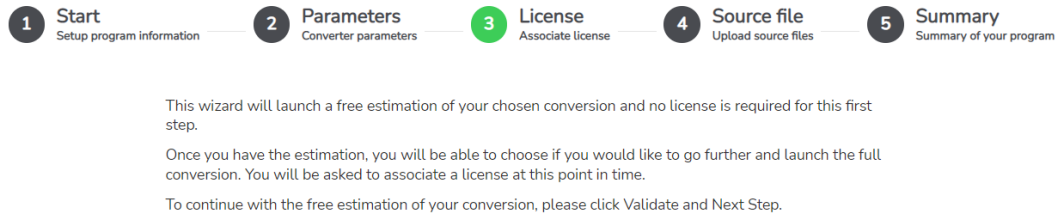
5. Select the PLC source brand: **SCADAPack Workbench Target 5**.

6. Select the PLC destination brand: **RemoteConnect Generator**.
7. Select the desired PLC destination CPU Type: the desired destination SCADAPack x70 model.
8. Click **Validate and Next Step**.



As only an evaluation of the logic is performed at this point, the License step does not apply.

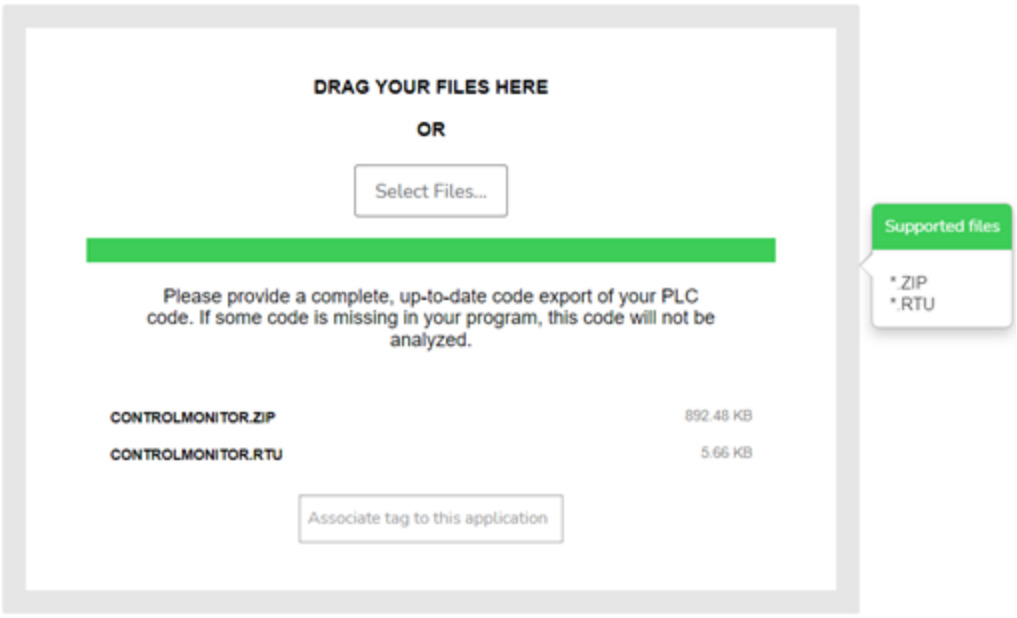
9. Click **Validate and Next Step**.



The source file to be uploaded is a *.zip file containing the entire saved Workbench Target 5 application. Most importantly, it needs to include the *.mdb database file.

A SCADAPack E Configurator *.rtu file, if available, may also be uploaded along with the Workbench *.zip file. The conversion tool can use this file to configure any DNP3 points or Modbus registers used in the project. Other communication and protocol settings need to be manually configured after the RemoteConnect application has been created. If this file is not available, DNP3 points or Modbus registers need to be configured later in RemoteConnect.

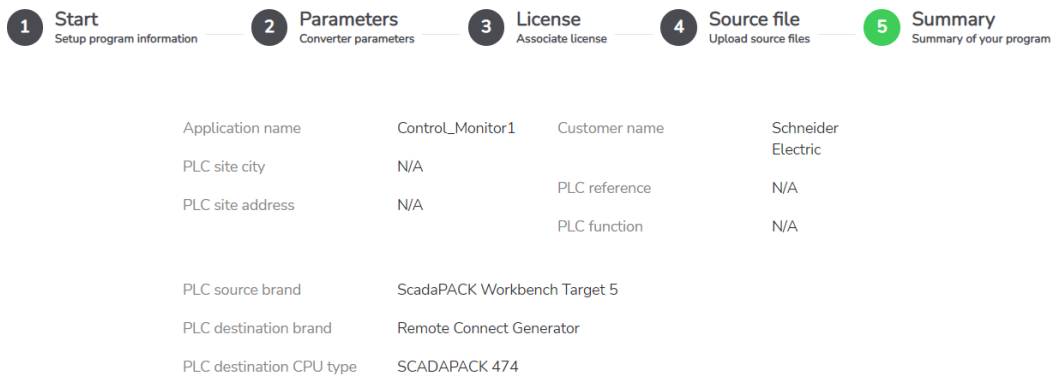
10. Either drag the file(s) onto the **Drag Your Files Here** area, or click **Select Files** and navigate to the desired file(s). After uploading the files, their names will appear as shown below.



When dependencies have been created to application libraries in separate projects from the main project, the main project Windows folder and each library folder (each including its *.mdb files) must be zipped and uploaded. However, only the main project to be converted is selected. See Importing the Converted Application 56.

11. When you are satisfied with the selection of uploaded files, click **Validate and Next Step** to continue.

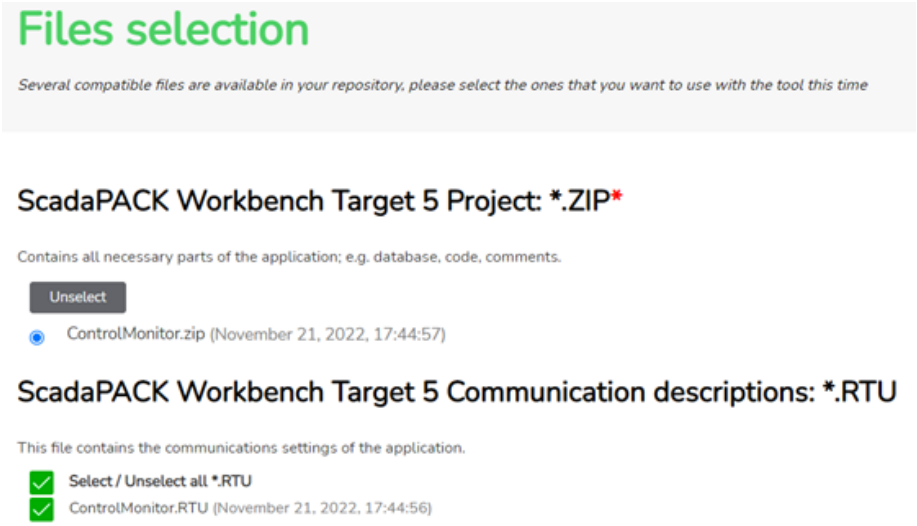
    A summary of the conversion to be performed is displayed.



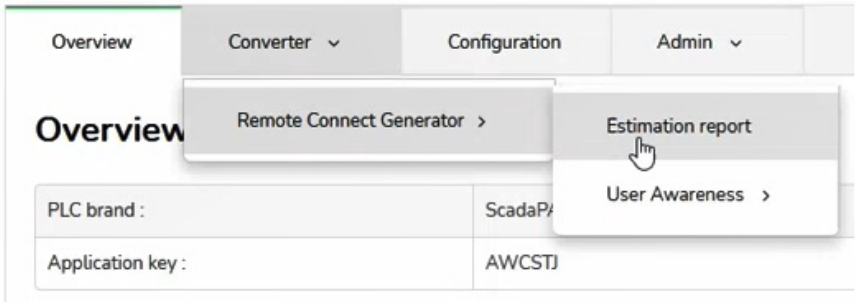
12. Review the summary details:

    - If changes are required, click **Previous Step.**
    - If the summary is acceptable, click **Launch** to continue.

13. If an *.rtu file was included along with the *.zip file in the previous step, a **Files selection** page is displayed. There are no changes required on this page. If you see this page, click **Launch** to continue.

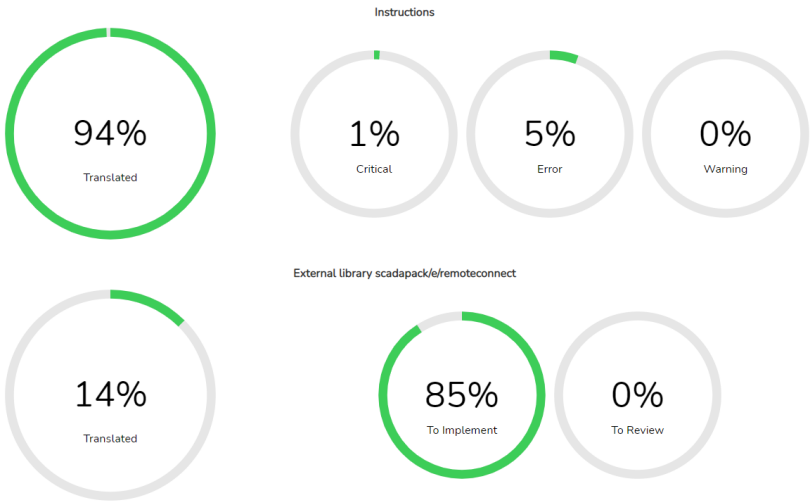


14. Wait for the **Analysis finished successfully** dialog to be displayed. Then click **Refresh** to continue.
15. The next page includes the **Overview** tab and several others. You can scroll down to view items such as **Key Information**, including number of instructions, lines of code to be converted, and variables to be converted.
16. Select **Converter** > **RemoteConnect Generator > Estimation report**.

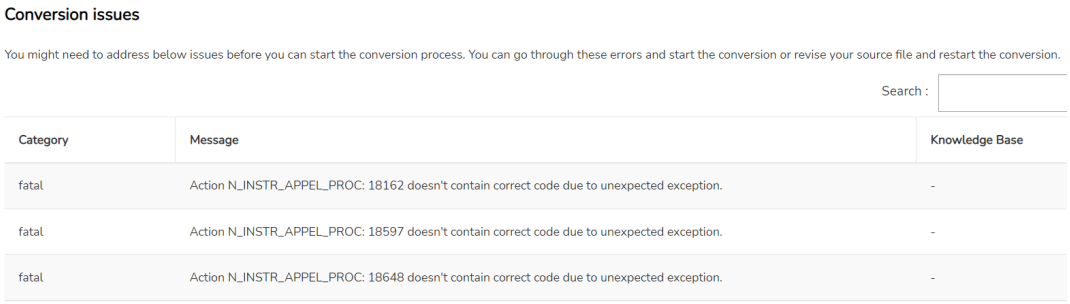


The Estimation Report provides information about how completely the conversion will be performed. There is no cost at this point, and you can exit the process, if needed.

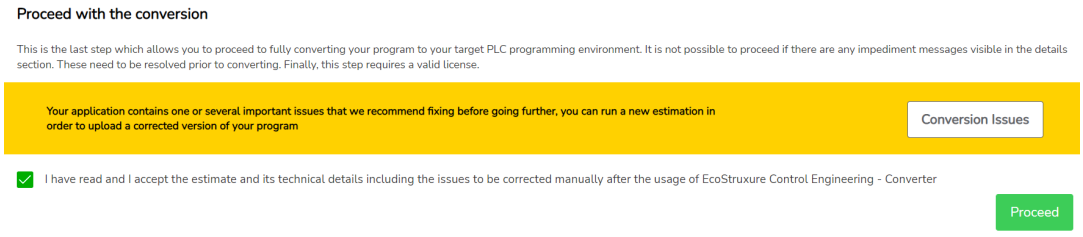
There are two groups of estimations. The first shows percentage of logic instructions that will be successfully converted, while the second shows percentage of library objects which will be successfully converted.



Any issues with conversion which are listed below as “fatal” or “critical” will need to be manually corrected after the project has been imported.



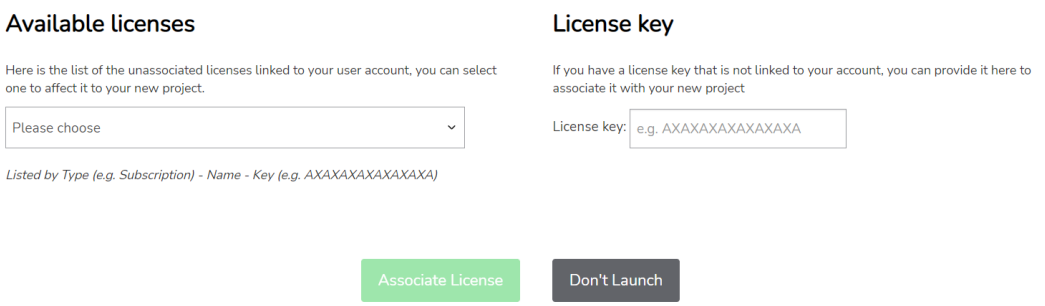
17. If you think that the errors can be corrected later in RemoteConnect, the next step is to agree to proceed regardless of reported issues or errors. Click the check box then click **Proceed**.



The license acquired earlier will be provided to perform the conversion.

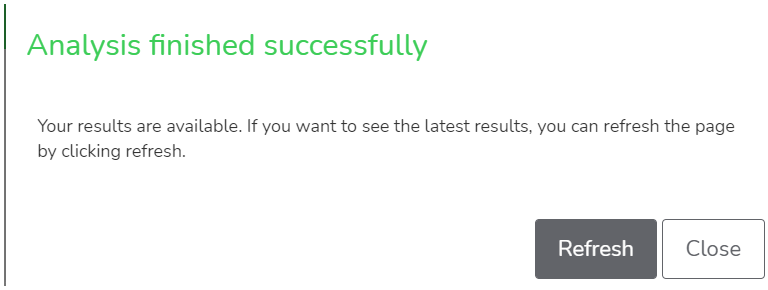
The license list appears in the drop-down list on the left side. However, if the license key was only acquired very recently it may not yet appear.

18. Select the license to be used, or copy and paste it as required, into the box on the right side.
19. When ready, click **Associate License** to begin the process.



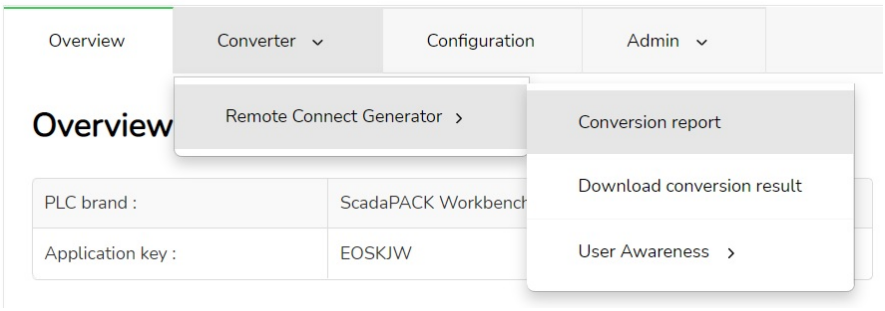
When the conversion is complete, a dialog is displayed.

20. Click **Refresh**.



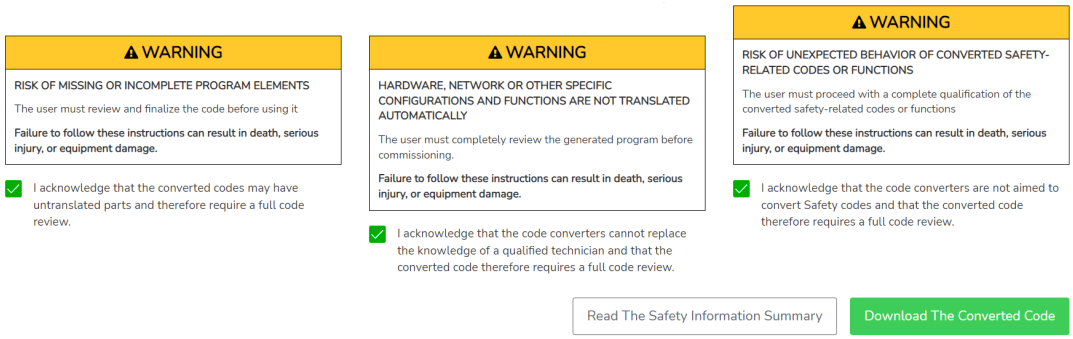
At this point, the license has been applied and the logic has been converted.

21. On the page that appears, select **Converter > RemoteConnect Generator > Conversion Report**.



A page similar to the **Estimation Report** is displayed.

22. Review the report and then scroll to the bottom of this page.
23. Review the Warning information provided, then read the **Safety Information Summary**.
24. After reviewing, click the three Warning check boxes, and then click **Download The Converted Code**.



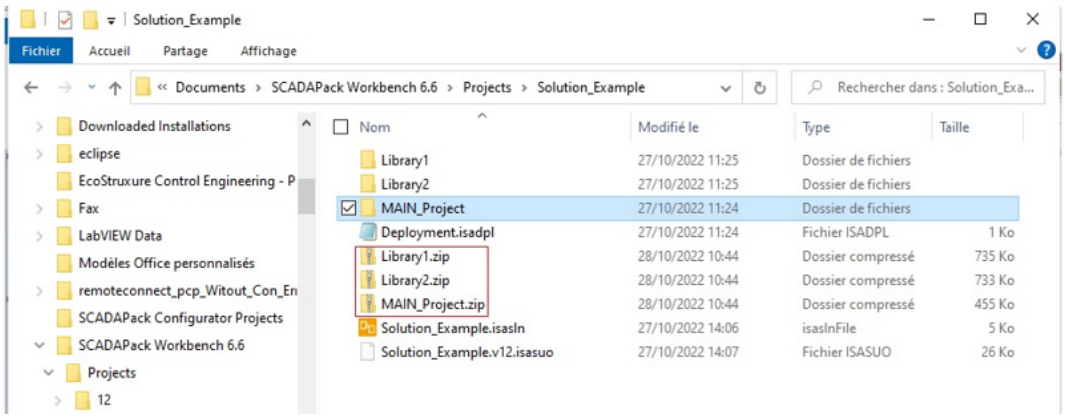
A zip file downloads. This contains two files: an .xef file and a .csv file.

The .xef file contains the converted logic to be imported into SCADAPack RemoteConnect. The .csv file contains the object database, which needs to be modified and then imported separately. See Importing the Converted Application 56.

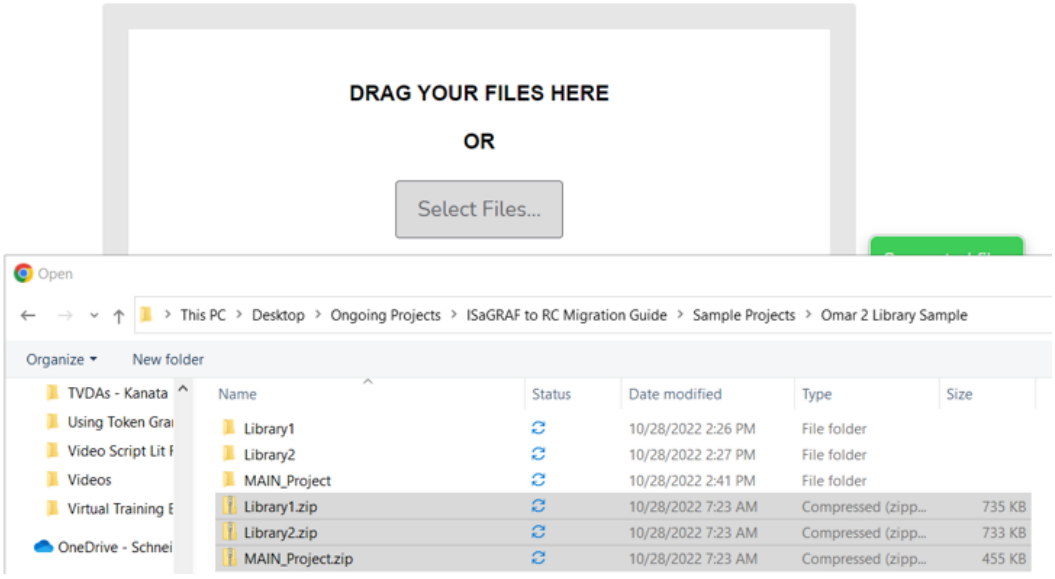
11.1.4 Converting a Project Containing Library Dependencies

The process of converting a project containing library dependencies is similar to performing a typical conversion. However, the Windows folders containing any libraries also used in the application need to be zipped separately. Both the main application zip file and the library zip file(s) need to be uploaded to the conversion tool at the same time. The brief example below covers the differences that will be seen.

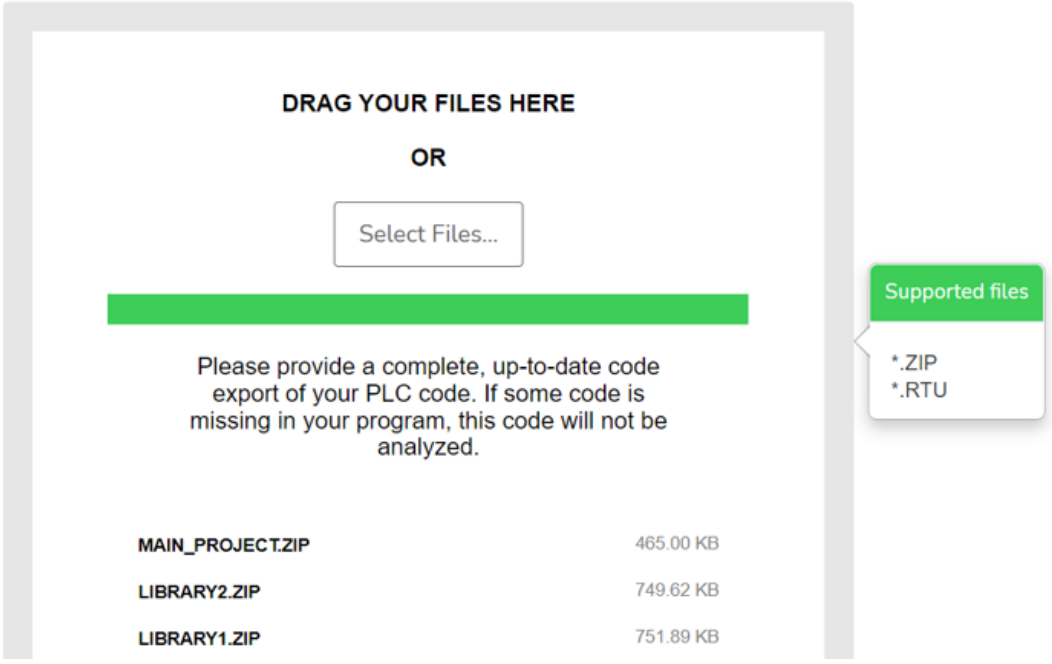
1. Using Windows Explorer, navigate to the main project application folder containing the *.mdb file.
2. Right-click on the main project application folder and select **Send To > Compressed (zipped) Folder**.
3. Navigate to the Windows folder containing the library that has been associated to the project as a dependency, and which contains an *.mdb file.
4. Right-click on the folder with the *.mdb file and select **Send To > Compressed (zipped) Folder**.
5. If more than one library has been associated as a dependency, create a zip file of each.



6. At the **Drag Your Files Here or Select Files** step in the conversion process, drag and drop or select the main project zip file and the required library zip files.



After uploading has completed, the zip files will be listed in the dialog. If available, a SCADAPack E Configurator *.rtu file may also be included.



7. In the next step, select the main project zip file.

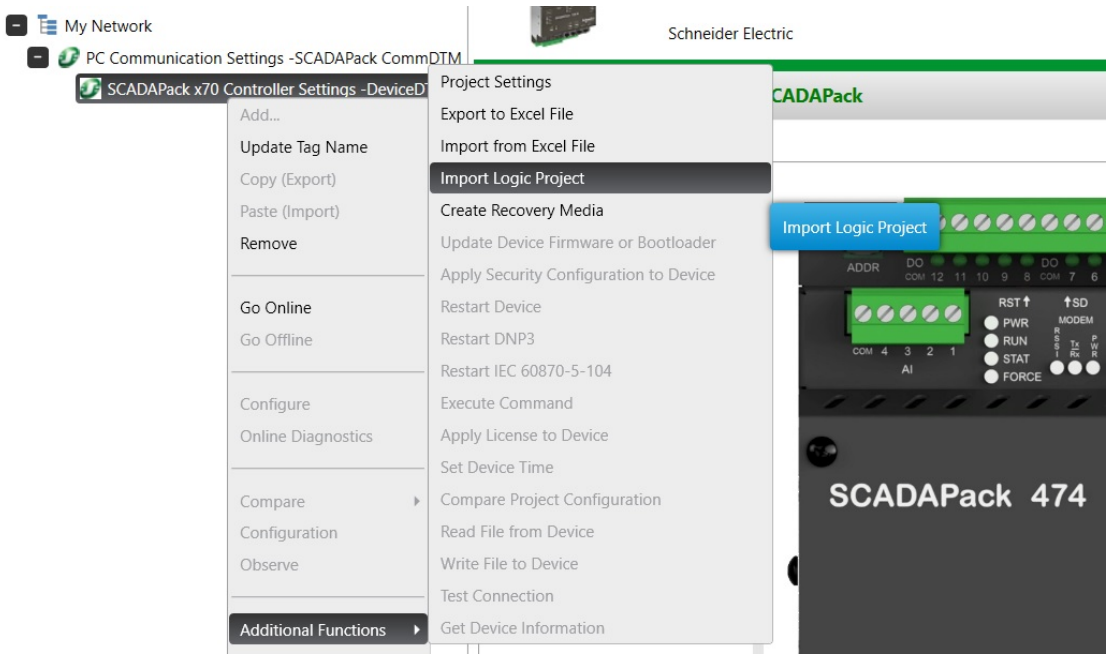
   The conversion will use the uploaded files, but it needs to know which file contains the main project.

The remaining steps of the conversion process will be the same as with the Typical conversion described in Performing a Typical Application Conversion 48.

11.2 Importing the Converted Application

The first step is to create a new project in SCADAPack RemoteConnect, specifying the desired SCADAPack x70 RTU type and communication features. Importing the converted logic is quite straightforward. Importing the object database is somewhat more challenging, but will be shown in detail.

1. Open SCADAPack RemoteConnect and click **Create Project**.
2. From the **SCADAPack Device Type** drop-down list, select the required device, and then select any required optional features such as communication options.
3. Click **Next**, **Next**, and **Finish** to create the application.
4. In the **My Network** area of SCADAPack RemoteConnect, right-click on the **SCADAPack x70 Controller Settings -DeviceDTM** and select **Additional Functions > Import Logic Project**.
5. Browse to the location of the .xef file created by the conversion tool.



6. Select the .xef file, click **Open** and then **OK** to import the file.
7. Save the SCADAPack RemoteConnect application after the import has completed.

   You need to separately import the Workbench project's object database which is contained in the .csv file provided in the converted download zip file. To begin, a spreadsheet containing the default object database will be created. The Workbench database will be added to the Objects tab of this spreadsheet, then imported into RemoteConnect.

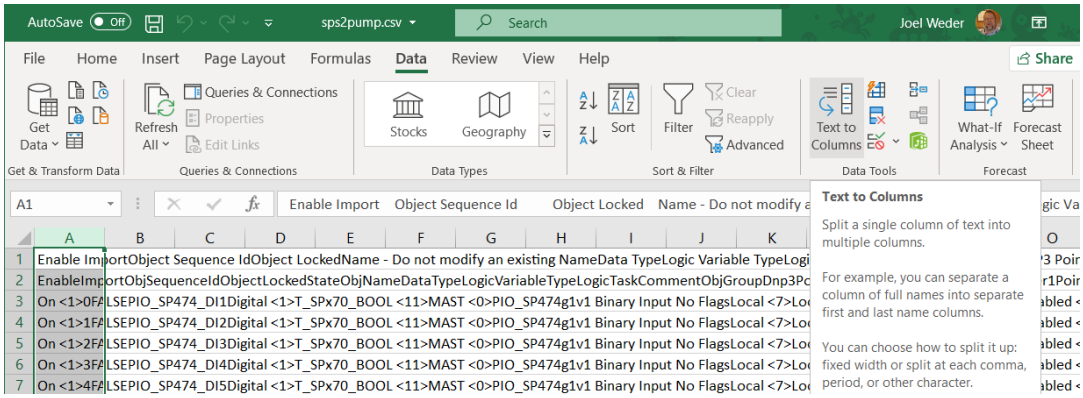
8. In the **My Network** area of RemoteConnect, right-click **SCADAPack x70 Controller Settings -DeviceDTM** and select **Additional Functions > Export to Excel File**.
9. Specify a filename and location, then save this temporary file.

   This saved Excel file includes multiple tabs, and contains the default project's configuration of physical I/O, communication and project settings. The process documented here replaces the existing (default) Objects list with the converted project's variables, as stored in the .csv file. Do NOT edit the parameters on any tab other than the Objects tab.

10. Open the .csv file which was provided as part of the conversion process.

    The file was created in tab-delimited format, with all source columns stored within column A. This format needs to be changed so that each category is in a separate column.

11. In Excel, click at the top of column A to select the entire column.
12. Go to the **Data** menu and then select **Text to Columns**.



13. Click **Finish**.

    Excel sorts the values into separate columns based on the original tab separation.

14. Save this modified .csv file.

15. Click in the upper-left corner of the spreadsheet to select ALL columns and rows (just above row 1 and to the left of column A), then copy the contents.

16. Navigate to the Excel (.xls) database file just exported from SCADAPack RemoteConnect and open it.

17. Select the tab **(2) Objects**.

18. Click in the upper-left corner of the **(2) Objects** spreadsheet, to select all columns and rows, then paste the copied information from the .csv file here.

19. Save the .xls file and close it.

20. In RemoteConnect, right-click **SCADAPack x70 Controller Settings -DeviceDTM** and select **Additional Functions > Import from Excel File**.

21. Navigate to and select the Excel database file exported earlier, which now contains a modified Objects tab.

22. Click **OK** and wait for the file to be imported, overwriting the defaults.

23. When the import has completed, save the RemoteConnect application.

24. In RemoteConnect select the **Objects** tab and verify that all of the objects from the original project have been successfully imported.

25. For each object imported from the Workbench Target 5 project which was tied to physical I/O, you need to connect these objects to RemoteConnect physical I/O objects.

    a. On the **Configuration** tab, expand **Physical I/O** and select **Local**.

    b. Select the configured SCADAPack x70 device. For example, SCADAPack 474.

    c. In the **Channel Configuration** area to the right, double-click on each physical I/O object in turn, e.g. DI1, to open its **Object Associations** dialog.

    d. Select **Associate with existing object**, then from the list choose the imported object that is to be associated with this physical I/O point.

e. Continue until all objects which were tied to physical I/O in the Workbench application have been associated with RemoteConnect physical I/O.

26. If any additional physical I/O hardware is required to provide equivalent I/O to the original RTU:

    a. On the **Configuration** tab, expand **Physical I/O** and select **Local**.

    b. Click **Add I/O** and choose an appropriate module.

    c. Enter a module name if desired, and an address, which begins at 0 for each type.

    d. Select the option **Manually connect existing objects to the I/O channels**.

    e. Select the new module, then in the **Channel Configuration** list double-click on each channel and **Associate** it with an existing (imported) object.

27. Save the RemoteConnect project when done.

11.3 Verifying Application Functionality

It is your responsibility to verify application functionality before placing the RTU into operation. This requires a full review of logic and configuration items, as well as testing of the functionality in field operation.

This document is not able to fully describe how to analyze the converted logic and object database.

Following are some suggestions:

1. Review items on the **Configuration** tab in SCADAPack RemoteConnect, checking for any incorrect settings in SCADAPack settings, Physical I/O or any of the communication parameters. Place some focus on the protocol in use e.g., Modbus™ or DNP3.

2. Review database objects on the **Objects** tab and verify that all objects have been imported. Open objects by double-clicking on them to check the data type and logic variable type and verify that any objects requiring Modbus or DNP3 addresses have been assigned properly.

3. With analog objects, if scaling is required or alert notifications (“level exceeded”) are desired, these may be added. If existing logic handles these functions this may not be required, though may allow removal of a significant amount of old logic.

4. In the Logic Editor, verify that all required global Elementary and Derived variables have been properly created. Some complex derived variables may need to be re-created manually.

5. Derived Function Blocks created in ISaGRAF 3.55 or Workbench may not have been converted properly. Check these carefully and verify.

6. Check that all logic sections have been re-created in Logic Editor, and that each includes the correct local variables.

7. View each logic section to verify that logic has been converted correctly.

8. When any required changes and/or corrections have been made, save the SCADAPack RemoteConnect project.

NOTE: In the current version of the logic conversion process, logic sections originally created with Function Block Diagram are converted to Ladder Diagram. Logic sections created with Structured Text remain as Structured Text, and logic sections created as Ladder Diagram remain as Ladder Diagram.

9. Write the configuration to the RTU, place it into Run mode, then perform a full re-commissioning of the facility.
10. If any further changes are required to restore original functionality, save and archive the modified application.

12 SCADAPack Workbench Differences

The following sections detail the differences between ISaGRAF 3.55 and Workbench Target 5 on the SCADAPack E controller. You may want to port existing ISaGRAF applications and there are differences that you need to address.

Import conversion tool

There is no import or conversion tool for converting ISaGRAF 3.55 applications to Workbench Target 5 available. Users with existing applications are required to re-enter the application using SCADAPack Workbench, as discussed in previous topics of this document.

The tool for conversion of Workbench Target 5 applications to SCADAPack RemoteConnect requires an upload to Schneider Electric's EcoStruxure Control Engineering Converter website, as discussed in Converting Workbench Application to RemoteConnect 46.

See:

- I/O Devices 61
- Programming Differences 65
- Standard SCADAPack Target 3 Function Blocks 66
- E Series SCADAPack Target 3 Function Blocks 71
- Online Modification 75
- Acquiring SCADAPack Workbench 76

12.1 I/O Devices

I/O devices for Workbench Target 5 are different than I/O boards for ISaGRAF.

Analog I/O devices for Target 5 support only a single ISaGRAF variable type per board. The corresponding boards in ISaGRAF could connect to any type of analog variable. More types of boards are required. See I/O Boards with Mixed Analog Variables 65.

I/O devices for Target 5 support a variable number of connections. I/O boards in Target 3 were fixed size. I/O devices are sized so they can replace any I/O board. Fewer board variations are required, which simplifies the process of selecting an I/O device for the user.

I/O devices have new names that better describe the function of the device, based on feedback from customers and technical support.

The following table shows ISaGRAF 3.55 vs Workbench Target 5 I/O device equivalents.

Replace the I/O boards with I/O devices according to the following table.

Group	Target 3 Board or FB	Target 5 I/O Device
AB DF1	Master Function Block	DF1_BOOL_READ DF1_INT_READ DF1_REAL_READ

	Master Function Block	DF1_BOOL_WRITE DF1_INT_WRITE DF1_REAL_WRITE
Modbus RTU serial	Master Function Block	MBUS_BOOL_READ MBUS_INT_READ MBUS_UINT_READ MBUS_DINT_READ MBUS_REAL_READ
	Master Function Block	MBUS_BOOL_WRITE MBUS_INT_WRITE MBUS_UINT_WRITE MBUS_DINT_WRITE MBUS_REAL_WRITE MBUS_INT_WRITE_SINGLE MBUS_UINT_WRITE_SINGLE
Modbus/TCP	Masterip Function Block	MTCP_BOOL_READ MTCP_INT_READ MTCP_UINT_READ MTCP_DINT_READ MTCP_REAL_READ
	Masterip Function Block	MTCP_BOOL_WRITE MTCP_INT_WRITE MTCP_UINT_WRITE MTCP_DINT_WRITE MTCP_REAL_WRITE
Modbus RTU in TCP	Masterip Function Block	MRTUTCP_BOOL_READ MRTUTCP_INT_READ MRTUTCP_UINT_READ MRTUTCP_DINT_READ MRTUTCP_REAL_READ
	Masterip Function Block	MRTUTCP_BOOL_WRITE MRTUTCP_INT_WRITE MRTUTCP_UINT_WRITE

		MRTUTCP_DINT_WRITE MRTUTCP_REAL_WRITE
RTU Database I/O	ain5501 ain5502 ain5503 ain5504 ain5521 ain8pt	RTU_RAW_READ RTU_RAW_WRITE_INPUT RTU_ENG_READ RTU_ENG_WRITE_INPUT
	aout5301 aout5302 aout5304 aout2pt aout4pt spaout	RTU_RAW_WRITE RTU_RAW_READ_OUTPUT RTU_ENG_WRITE RTU_ENG_READ_OUTPUT
	cntr5410 cntrctrl cntrint dinctrl	RTU_COUNTER_READ RTU_COUNTER_WRITE_INPUT
	din5401 din5402 din5403 din5404 din5405 din5414 din5421 din8pt din16pt dinint dinoptsw	RTU_BIN_READ RTU_BIN_WRITE_INPUT
	dout5401	RTU_BIN_WRITE

	dout5402 dout5406 dout5407 dout5408 dout5409 dout5411 dout8pt dout16pt	RTU_BIN_READ_OUTPUT
SCADAPack I/O	sp100	SCADAPack Complex Equipment boards are not supported in Workbench Target 5. These need to be replaced with one or more RTU_ I/O boards, as described above in the row RTU Database I/O 63.
	sp31x	
	sp33x	
	sp350	
	ss4203dr	
	ss4203ds	
	sp5601	
	sp5602	
	sp5604	
	sp5606	
	sp5607	
	in5505	
	in5506	
	dout5415	

12.1.1 SCADAPack Complex Equipment

ISaGRAF Target 3 complex equipment objects are collections of I/O boards. The SCADAPack complex equipment objects are not supported in Workbench Target 5. You need to access each I/O type separately with RTU_ I/O boards as described in the I/O Devices 63 table.

12.1.2 I/O Boards with Mixed Analog Variables

A user application which connected more than one type of analog variable to an I/O board in Target 3 will replace the I/O board with multiple I/O devices. The number of I/O device channels will be set to match the number of consecutive points of one type from the Target 3 program.

This permits conversion of any I/O board configuration and may in some cases require one I/O device per variable.

12.1.3 Modbus I/O Devices

Modbus I/O devices provide the same functionality as in ISaGRAF Target 3. The changes in the I/O Devices required for Workbench Target 5 improve the user experience somewhat.

- The limitation of 100 I/O boards in ISaGRAF Target 3 is increased to 200. This expands the number of registers that can be polled.
- The number of points polled per device is now configurable. For most configurations, this reduces the number of I/O Devices needed.

12.2 Programming Differences

The main differences between ISaGRAF Target 3 and Workbench Target 5 programming languages are described below.

TSTART and TSTOP functions

The TSTART and TSTOP functions are not supported. These functions were available in ST language only.

The TON function block replaces these functions. ST programs need to be modified to use TON.

Variable types

For ISaGRAF Target 3 configurations, the following variable types are available:

- Boolean Variables (BOOL)
- Double Integer Variables (DINT)
- Real Variables (REAL)
- Timer Variables (TIME)
- String Variables (STRING)

For Workbench Target 5 configurations, the following variable types are available:

- Boolean Variables (BOOL)
- Short Integer Variables (SINT)
- Unsigned Short Integer (USINT) or BYTE
- Variables

- Integer Variables (INT)
- Unsigned Integer (UINT) or WORD Variables
- Double Integer Variables (DINT)
- Unsigned Double Integer (UDINT) or Double Word (DWORD) Variables
- Long Integer Variables (LINT)
- Unsigned Long Integer (ULINT) or Long Word (LWORD) Variables
- Real Variables (REAL)
- Long Real Variables (LREAL)
- Timer Variables (TIME)
- Date Variables (DATE)
- String Variables (STRING)

Operators

A number of operators are replaced. For example “&” is replaced with “AND”.

See the ISaGRAF documentation included with SCADAPack Workbench for documentation of operators.

12.3 Standard SCADAPack Target 3 Function Blocks

This section specifically discusses conversion from a Standard SCADAPack running a Target 3 application. Refer to E Series SCADAPack Target 3 Function Blocks 71 if converting from a Target 3 application running in an E Series Target 3 RTU.

See:

- Target 3 Functions and Function Blocks Not Supported in Target 5 66
- RemoteConnect Features Which May Replace Function Blocks 70

12.3.1 Target 3 Functions and Function Blocks Not Supported in Target 5

The following list includes functions and function blocks specifically created for the Standard Target 3 SCADAPack, which are not supported in Workbench Target 5. Some of these do have an equivalent in Target 5, but it will be necessary to manually re-create them. See RemoteConnect Features Which May Replace Function Blocks 70 for a list of function blocks which may be replaced if desired by configurable features in RemoteConnect.

Function or Function Block	Description
clearsf	Clear store and forward table
cominfo	Serial port status
ctlrstat	Get controller status code

Devconf	Set device configuration
dial	Control dialup modem
dlog	Data Logger Function Block
dlogcnfg	Data Logger Configuration Function Block
dlogf	Data log to file
dlogfcnfg	Configure datalogging to file
dlogread	Data Logger Extract Function Block
dnpconn	Get DNP connection status
dnpevent	Get DNP change event statistics
dnplog	Log a DNP event for a point
dnppoll	Trigger a DNP class poll of a slave device
dnpport	Get DNP diagnostic data for a port
dnpstn	Get DNP diagnostic data for a remote station
dnpsync	Trigger DNP clock synchronization to slave
dnpunsol	Trigger a DNP unsolicited response message
dtroff	Control DTR
flow	Flow Accumulator
forceled	Get force LED state
getclock	Get current date and time
getcom	Get serial port settings
Getftp	Get FTP server state
gethart	Get HART module configuration
getipi	Get Interface IP Address
getled	Get LED power settings

getmbip	Get Modbus IP Protocol Settings
getmtcp	Get Modbus/TCP Settings
getmtcpi	Get Modbus/TCP Interface
getmtpi2	Get Modbus/TCP Interface method 2
getpmode	Get controller power mode
getprot	Get protocol settings
getprot2	Get protocol settings method 2
getregb	Get value of boolean register
getregf	Get value of floating point register
getregsl	Get value of signed long integer register
getregss	Get value of signed short integer register
getregus	Get value of unsigned short integer register
getsf	Get store and forward entry
getsfip	Get Store and Forward Entry
getsfip2	Get Store and Forward Entry method 2
hart0	Send HART command 0
hart1	Send HART command 1
hart2	Send HART command 2
hart3	Send HART command 3
hart33	Send HART command 33
inimodem	Initialize modem
ipstatus	Summary of IP Connections
master	Send Modbus master command
masterip	Send Modbus IP Master Command

mbusinfo	Modbus protocol status
pida	Analog Output PID Function Block
pidd	Discrete Output PID Function Block
protinfo	Protocol status
rxstring	Receive a Message String
setclock	Set current date and time
setcom	Set serial port settings
Setftp	Set FTP server state
sethart	Set HART module configuration
setipi	Set Interface IP Address
setled	Set LED power settings
setmbip	Set Modbus IP Protocol Settings
setmtcp	Set Modbus/TCP Settings
setmtcpi	Set Modbus/TCP Interface
setmtpi2	Set Modbus/TCP Interface method 2
setpmode	Set controller power mode
setprot	Set protocol settings
setprot2	Set protocol settings method 2
setregb	Set value of boolean register
setregf	Set value of floating point register
setregsl	Set value of signed long integer register
setregss	Set value of signed short integer register
setregus	Set value of unsigned short integer register
setResp	Control Modbus Exception Response

setsf	Set store and forward entry
setsfip	Set Store and Forward Entry
setsfip2	Set Store and Forward Entry method 2
sleep	Put controller in sleep mode
toeeprom	Save settings to EEPROM
total	Non-Volatile Totalizer
txstring	Transmit a Message String

12.3.2 RemoteConnect Features Which May Replace Function Blocks

Certain Target 3 function blocks may not need to be re-created in Workbench Target 5 before conversion to RemoteConnect, as similar functionality is provided in RemoteConnect with configuration rather than with function blocks. Where available, however, logic function blocks may be used if preferred. Features configurable in RemoteConnect include, but are not limited to:

RemoteConnect Feature	Target 3 Function Block
Data Logging	dlogf
	dlogfcnfg
Modbus Scanner	master
	masterip
Modbus Store and Forward	clearsf
	setsf
	setsfip
DNP3 Configuration	dnplog
	dnppoll
Serial Communication	setcom
IP Communication	setipi
	setmbip

	setmtcp
	setmtcpi

12.4 E Series SCADAPack Target 3 Function Blocks

This section specifically discusses conversion from an E Series SCADAPack running a Target 3 application. Refer to Standard SCADAPack Target 3 Function Blocks 66 if converting from a Target 3 application running in a standard model SCADAPack.

Custom Functions and Function Blocks from E Series ISaGRAF Target 3 are supported in SCADAPack Workbench Target 5, with some exceptions.

See:

- Supported Functions and Function Blocks 71
- Operate Function 74
- Superseded Peer Communication Function Blocks 74
- Peer Queued Function Blocks 75
- Obsolete Function Blocks 75

12.4.1 Supported Functions and Function Blocks

The E Series Target 3 functions and function blocks supported in Workbench Target 5 are shown in the following table.

Type	**Name** (variants not listed)
Point Attribute Function Blocks	getpnt setpnt rdfld rdrec rdstring setatr
Real Time Clock Function Blocks	os_time loc_time timedate set_timedate
DNP3 Peer Communication Function Blocks	dc_poll

	peer_rdq peer_wrq peer_rdx peer_wrx peer_rdc peer_wrc
Digital Output Controls	rtucrob
Modbus Master and Client Function Blocks	modbus_setup_rd modbus_setup_wr
PLC Communications Control Function Blocks	mbusctrl mtcpctrl df1ctrl
Serial Port User Communication Functions	comopen comclose comrx comrxb comrxclr comtx comtxb getport setport
Process Function Blocks	pida pidd total flow
Miscellaneous Function Blocks	gen_evt genmsevt ana_time cmd_exec rtuparam

	chgroute chgrtnum chgrtprt
TCP/IP Interface Functions	ip_add ip_del ip_cycgw ip_ping string_to_ip ppp_echo
Alarm Group Functions	almadd almproc almload almclr
File System Management Functions	f_del f_deltree f_copy f_join f_ren f_mkdir f_rmdir f_cd f_dsksel f_pwd f_dv_rdy
Directory Information Function Blocks	findfile dir_info
File Read/Write Functions	f_wopen f_ropen f_close f_eof

	fa_read fa_write fm_read fm_write
AGA Gas Calculation Function Blocks	aga8dtl aga8grs aga3cfac aga3orif aga7_9
Rod Pump Controller Function Blocks	rpc_sampler license_check_rpc
Progressive Cavity Pump Function Blocks	license_check_pcp
HART MTL4851 Function Blocks	hart_device_scan hart_mtl4851_ctrl hart_cmd

12.4.2 Operate Function

The OPERATE function in ISaGRAF Target 3 was a standard function with an OEM-specific implementation. It is not supported in Workbench Target 5.

It was primarily used to allow a Target 3 application to write a point database value associated with an ISaGRAF input variable (e.g., pre-set or override the database value). A workaround is to use the point database write function block (setpntxx).

12.4.3 Superseded Peer Communication Function Blocks

These function blocks are not supported:

- RDxxBIN
- RDxxANA
- RDxxFLT
- WRxxBIN
- WRxxANA
- WRxxFLT

12.4.4 Peer Queued Function Blocks

The peer-to-peer functionality adopts the use of the newer function blocks defined in Supported Functions and Function Blocks 71, replacing the use of function blocks listed in Superseded Peer Communication Function Blocks 74.

Confirm the correct operation of peer-to-peer communications, especially when each peer of a communications link is of a differing RTU type and / or is of a differing Target version. For each RTU and software version review its corresponding DNP3 Interoperability Document to verify matching specifications for data transfer.

Ultimately, as part of a comprehensive set of test and verification procedures, the peer-to-peer communications should be tested and reviewed once converted.

12.4.5 Obsolete Function Blocks

These function blocks are not supported. They are superseded by rtucrob.

- rtupulse
- rtupuls2

This conversion function is not supported. It is superseded by the native ANY_TO_STRING function.

- rea_msg

This function block is not supported as the corresponding I/O devices are not supported.

- koyoctrl

12.5 Online Modification

Online program modification is improved with Workbench Target 5. Online modification is allowed for code sequences and parameters of I/O devices. And On-line modification to user parameters on I/O devices is allowed. However, it is not possible to modify variable definitions, application parameters, or I/O connections. Workbench documentation provides the following information:

You can modify a device while it runs. This is sometimes necessary for chemical processes where any interruption may jeopardize production or safety. When performing online changes, you can choose to update a running device at the time of download or at a later time. However, online changes should be used with care. Workbench may not detect all possible conflicts generated by user-defined operations as a result of these online changes.

The initial values of variables are applied upon starting devices. Online changes do not start devices. When performing online changes, you can only modify code sequences: you cannot modify variable definitions, application parameters, and I/O connections.

Workbench compares the symbol table checksum to detect changes to any variable, program, or SFC element name. If a step is active when the switch occurs, its non-stored (N) actions are lost. The new step activation actions are not executed. Actions executed at the deactivation of the step are those carried over to the new application code. If a transition is valid when the

switch occurs, its receptivity equation is updated. The new downloaded application code is not backed up on the PLC. The backup is the version that was previously downloaded using standard download commands.

To support online changes, the Target PLC must have sufficient free memory space to enable the storage of the modified version of the application code. Both versions of the application code are stored in the PLC memory during the switch operation.

…

The online change procedure ensures that the Target PLC always maintains a complete and reliable running application while enabling the user to control the timing of the sample operations in a safe and efficient manner. The procedure also enables modifying the project as often as possible. Regardless of the process itself, the online change is essentially the same as performing stop, start, and download operations where the only differences are no loss of variable states and a switching time running across one or two cycles. During a switch between the previous and current applications, all internal, input, or output variables retain the same values before and after the application modification while SFC actions are not executed and token movements are suspended.

12.6 Acquiring SCADAPack Workbench

If you do not have a copy of SCADAPack Workbench or SCADAPack E Workbench, contact you sales channel to acquire the software.